

SOLO NUESTROS "QL" SE EXPLICAN EN ESPAÑOL
DOCUMENTACION EN ESPAÑOL, CONTENIENDO:
• INTRODUCCION • GUIA DEL PRINCIPIANTE • MANUAL DEL SUPERBASIC
• MANUAL TECNICO DE REFERENCIA
• APLICACION "QL" QUILL • APLICACION "QL" ABACUS • APLICACION "QL" ARCHIVE
• APLICACION "QL" EASEL
MICRO WORLD
Modesto Lafuente, 63 Telf. 253 94 54 28003 MADRID
Colombia, 39-41 Telf. 458 61 71 28016 MADRID
José Ortega y Gasset, 21 Telf. 411 28 50 28006 MADRID
Fuencarral, 100 Telf. 221 23 62 28004 MADRID
Padre Damián, 18 Telf. 259 86 13 28036 MADRID
Avda. Gaudí, 15 Telf. 256 19 14 08015 BARCELONA
Ezequiel González, 28 Telf. 43 68 65 40002 SEGOVIA
Stuart, 7 Telf. 891 70 36 ARANJUEZ (Madrid)
MICROHOBBY SEMANAL
REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR
AÑO II- N.º 11
95 PTAS.
EDITA HOBBY PRESS, S.A.
Canarias 105 ptas.
NOVEDAD
MUGSY: EL REY DEL HAMPA EN VIDEOCOMIC
SOFTWARE
DISEÑA TUS PROPIOS GRAFICOS
UTILIDADES
RUTINA EN CODIGO MAQUINA DE CARGA Y GRABACION ¡CON VELOCIDAD VARIABLE!
PROGRAMAS
EL DEFENSOR
MISION LUNAR
EL BARON ROJO
HOP
¡REGALAMOS UN 'QL' CADA MES!

Franqueo
Postal

HOBBY PRESS, S. A.

Apartado de Correos

n.º **54.062** (Apartados Altos)

MADRID

Director Editorial
José I. Gómez-Centurión
Director Ejecutivo
Domingo Gómez
Redactor Jefe
Africa Pérez Tolosa
Diseño
Jesús Iniesta
Maqueta
Rosa M. Capitel
Redacción
José María Díaz
Gabriel Nieto
Colaboradores
Jesús Alonso, Lorenzo Cebeira, Primitivo de Francisco, Rafael Prades, Víctor Prieto
Fotografía
Javier Martínez
Carlos Candel
Portada
José María Ponce
Dibujos
Manuel Berrocal, J.R. Ballesteros, A. Perera, F.L. Frontán, J. Septien, J.M. López Moreno

Edita
HOBBY PRESS, S.A.

Presidente
María Andrino
Consejero Delegado
José I. Gómez-Centurión
Administrador General
Ernesto Marco

Jefe de Publicidad
Marisa Esteban
Secretaria de Publicidad
Concha Gutiérrez
Publicidad Barcelona
Isidro Iglesias
Tel.: (93) 307 11 13

Secretaria de Dirección
Marisa Cogorro

Suscripciones
M.ª Rosa González
M.ª del Mar Calzada

Redacción, Administración y Publicidad
Arzobispo Morcillo, 24, oficina 4.
28029 Madrid
Telf.: 733 50 12

Distribución
Coedis, S.A. Valencia, 245.
Barcelona.

Imprime
Rotedic, S.A.
Carretera de Irún, Km. 12,450
Tel.: 734 15 00
Fotocomposición
Consulgraf
Nicolás Morales, 34 - 1.º
Tel.: 471 29 08
Fotomecánica
Zescán
Nicolás Morales, 38
Tel.: 472 38 58
Depósito Legal:
M-36.598-1984

Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América, 1.532. Telf.: 21 24 64. 1209 BUENOS AIRES (Argentina).

Derechos Exclusivos
«Sinclair Users», «Sinclair Programs» y «Sinclair Projects» de EMAP Publications (Londres).

Se solicitará control OJD

MICROHOBBY ESTA SEMANA

Año II · N.º 11 · 15 al 21 de enero de 1985
95 ptas. (Sobretasa Canarias 10 ptas.)

TRUCOS

CONVERSOR DECIMAL-BINARIO

Todos aquellos usuarios del Spectrum que además tengan el «gusanillo» de la programación, se habrán encontrado, en más de una ocasión, con la necesidad de un programa de utilidad que les presentará los equivalentes en hexadecimal y/o binario de un número decimal.

En el caso de la equivalencia en hexadecimal, realizado en Basic no presenta ningún problema, de hecho, ya hemos publicado un programa que lo hace.

Aunque para el caso binario el algoritmo de programación es muy sencillo, tiene el inconveniente de la velocidad; es muy lento.

Por tanto hemos construido una pequeña subrutina en lenguaje máquina que responde instantáneamente en cuanto le damos el valor decimal. Además, el código máquina está ensamblado en la zona de la memoria intermedia de la impresora, por lo que «no ocupa memoria»; no obstante, esta práctica tiene sus inconvenientes: no podemos utilizar las sentencias Basic que involucran a la impresora.

En conjunción con el programa Basic que obtiene el valor decimal, puede emplearse como subrutina en cualquiera de nuestros programas.

```
   1 DATA 253,203,2,134,237,75,0,
91,197,205,43,45,205,227,45,209,
62,6,215,62,8,215,74,205,41,91,6
2,32,215,75,205,41,91,62,13,215,
201,6,8,203,33,62,49,56,2,62,48,
215,16,245,201
  10 FOR i=23300 TO 23300+50: RE
AD x: POKE i,x: NEXT i
  15 INPUT "VALOR ";V: LET H=INT
 (V/256): LET L=V-256*H: POKE 23
296,L: POKE 23297,H
  20 IF NOT v THEN STOP
  25 RANDOMIZE USR 23300: GO TO
15
```

EFECTOS MUSICALES

Uno de nuestros lectores, Luis García Lleo, nos envía el siguiente programa que permite obtener de nuestro ordenador, el famoso «efecto de trémolo» a pesar de la exigua capacidad sonora del Spectrum.

```
  10 REM *** Efecto de tremolo *
**
  20 FOR a=1 TO 4
  30 FOR b=1 TO 50
  40 BEEP .05,5
  50 BEEP .05,5+a
  60 NEXT b
  70 PAUSE 25
  80 NEXT a
```

CASTELLANIZAR SPECTRUM



Para poder tener en nuestro ordenador al menos una de las letras propias de nuestro idioma, teclée el siguiente programa y luego pulse CAPS SHIFT + 9 + N.

```
10 POKE USR "n",BIN 00111100
```

PARA MANEJAR LINEAS LARGAS

En determinadas ocasiones, y a la hora de ahorrar memoria, conviene colocar más de un comando en una misma línea Basic, ya que el intérprete consume 5 bytes como mínimo para identificar unívocamente cada una de ellas (2 bytes como mínimo para el número de línea, 2 para la longitud de la línea y uno para el carácter marcador de fin de línea).

Sin embargo, como contrapartida a este ahorro de memoria, existe la dificultad inherente a editar líneas largas para su correción, sobre todo si dicha correción se encuentra al final de la línea (como sucede casi siempre, según la ley de máxima fatalidad).

Una de las soluciones se dió ya en esta misma sección de trucos, y consistía en aumentar la velocidad de repetición de las teclas. La otra que proponemos aquí es, simplemente, pulsar CAPS SHIFT + 7 (u 8) durante un breve tiempo y luego soltar la tecla CAPS; al seguir pulsando 7 u 8 el cursor se moverá a lo largo de la línea como si todavía estuvieran pulsadas ambas teclas.

En este espacio también tienen cabida los trucos que nuestros lectores quieran proponer.
Para ello, no tienen más que enviarlos por correo a MICROHOBBY, C/Arzobispo Morcillo, 24, of. 3 y 4, Madrid-28029.

MANEJO DE LOS GRAFICOS DEFINIDOS POR EL USUARIO

Jesús ALONSO

Para entender los «terribles» U.D.G. es necesario comprender primero la forma en la que trabaja la memoria del ordenador. Esperamos que tras la lectura de este artículo nadie tenga problemas en definir gráficos para sus propios programas.

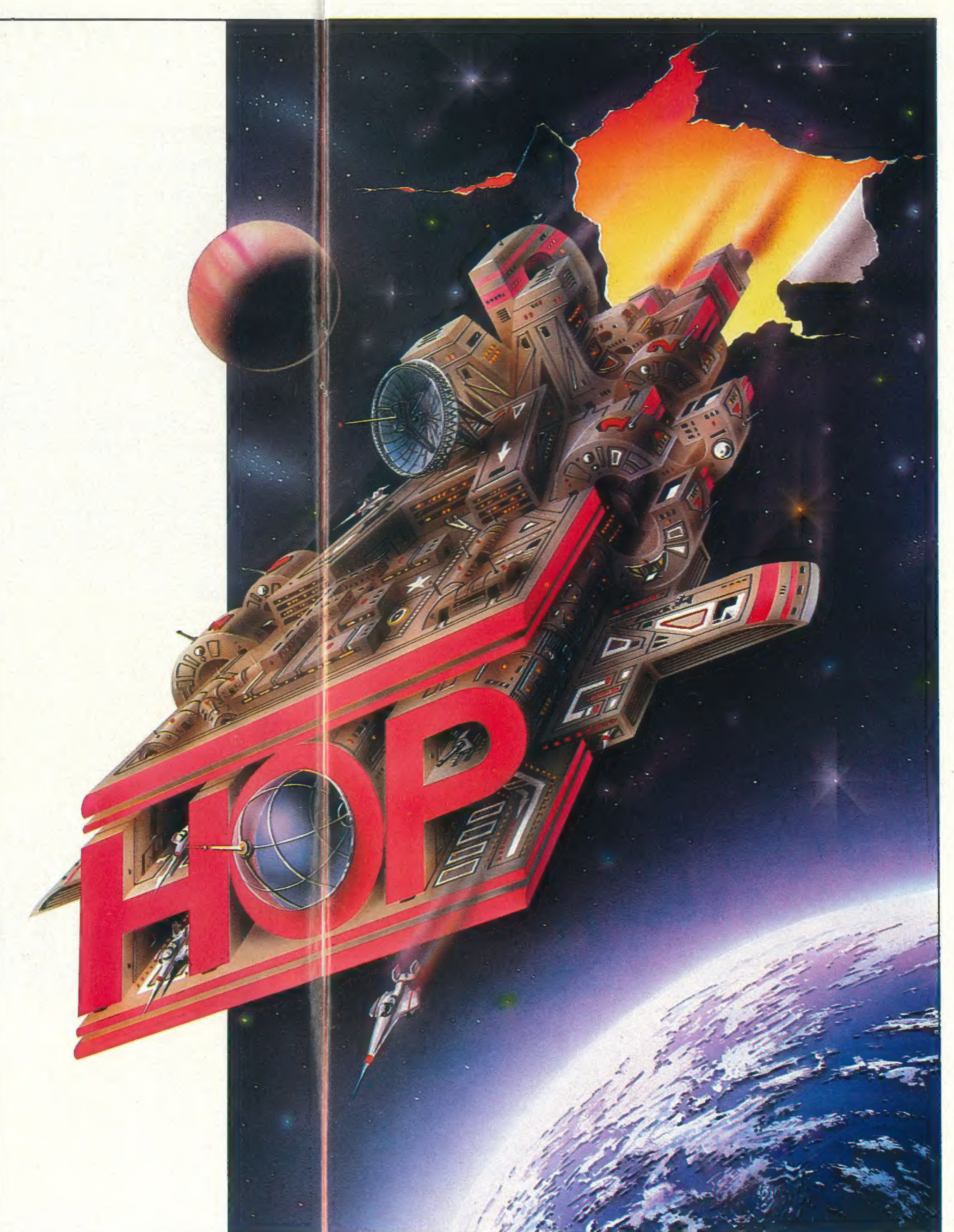
A juzgar por las cartas recibidas en nuestra redacción, parece ser que uno de los principales problemas con que se encuentra el joven programador que se está iniciando en el BASIC, es la definición de U.D.G. (Gráficos Definidos por el Usuario). Este problema se debe, en la mayoría de los casos, al desconocimiento sobre la forma en que trabaja la memoria del ordenador. Para definir gráficos es indispensable utilizar el comando POKE que trabaja directamente sobre la memoria. Un gran número de lectores nos preguntan, también sobre la forma de utilizar este comando. Vamos a intentar, con este artículo, dar una explicación exahustiva sobre la utilización del comando POKE y la definición de gráficos, para lo cual empezaremos por explicar cómo «cuenta» el ordenador.

Cómo «cuenta» el ordenador

Cuando nuestros remotos antepasados se plantearon el problema de numerar los objetos, decidieron en un principio, lo que parecía más evidente: utilizar los dedos de la mano. Hacían corresponder cada objeto con un dedo de la mano, y de esta forma, podían contar hasta cinco (de hecho nuestra palabra «contar» proviene de un vocablo griego que significa «hacer cinco»). Si utilizaban los dedos de ambas manos les era posible numerar un máximo de diez objetos. Posteriormente, los árabes idearon una serie de signos que se correspondían con los diez primeros números (del cero al nueve); el número con el que se correspondía cada signo venía dado por el número de ángulos que tuviera ese signo, así el cero era un círculo que, como sabemos, no tiene ningún ángulo; el uno era una figura con un ángulo; el dos tenía dos ángulos, y así sucesivamente hasta el nueve, que era una figura con nueve ángulos. Estos signos, ligeramente modificados, son los números que han llegado hasta nuestros días, por eso se les llama con frecuencia «Números arábigos».

Un sistema de numeración que sólo permita contar hasta nueve, no resulta demasiado útil, por otro lado no era cosa de inventar cien signos para contar hasta cien y luego pretender recordarlos todos, ¿se imagina el lector una figura con cien ángulos? Parecía mejor idea agrupar estos diez signos según una regla definida y obtener así cualquier número por grande que este fuera. Se pensó que los signos (también llamados dígitos) se colocarían en fila, el primero por la derecha representaría su valor multiplicado por uno, el segundo representaría su valor multiplicado por diez, el tercero, su valor por cien, y así sucesivamente, de esta forma el número 1984 significa: cuatro por uno más ocho por diez más nueve por cien más uno por mil. Pero uno, diez, cien y mil son potencias de diez (uno es diez elevado a cero, diez es diez elevado a uno, cien es diez elevado a dos, mil es diez elevado a tres, etc.), como se ve, los exponentes son los propios números, pero la base es siempre diez, por eso se llama a este sistema numeración decimal o de base diez. La base diez es tan común para nosotros que la creemos la única posible, pero de hecho se debe al accidente evolutivo de que tengamos diez dedos. No es la mejor base posible y existen otras. El BASIC es un lenguaje de programación ideado para que los humanos nos entendamos fácilmente con los ordenadores y por eso utiliza la base diez que es la más común para nosotros, pero el ordenador no tienen diez dedos y utiliza a nivel interno otra base de numeración que le resulta más fácil, la base dos.

Podemos imaginar la memoria de un ordenador como un tablero lleno de interruptores, cada uno de ellos puede estar «encendido» o «apagado». De forma arbitraria hacemos corresponder el estado «apagado» con el dígito «cero», y el estado «encendido» con el dígito «uno», con la posición de cada interruptor podemos numerar dos objetos, uno se llamaría «cero» y el otro se llamaría «uno», o lo que es lo mismo, podemos contar desde cero hasta uno. De nuevo, no resulta muy útil un sistema de numeración que sólo permita contar hasta uno. Vamos a hacer con nuestros interruptores lo mismo que los árabes con sus signos, vamos a agrupar nuestros interruptores. Supongamos que los agrupamos en filas de ocho interruptores (se pueden agrupar de cualquier otra forma, pero nuestro ordenador los agrupa de ocho en ocho). A cada interruptor lo llamamos «bit» y puede valer «cero» o «uno» según esté apagado o encendido, y al conjunto de ocho interruptores le llamamos «byte» o posición de memoria (a veces se le llama «palabra»), de esta forma estamos trabajando con «bytes» de ocho «bits» (los informáticos dicen «palabras de ocho bits»). El primer interruptor (bit) por la derecha, valdrá su valor multiplicado por uno (dos elevado a cero); el segundo, su valor multiplicado por dos (dos elevado a uno); el tercero, su valor multiplicado por cuatro (dos elevado a dos), y así hasta el último, que valdrá su valor multiplicado por 128 (dos elevado a siete). Vamos a situar nuestros ocho interruptores de forma que estén todos «encendidos». Nuestra posición de memoria contendrá el número 11111111 en binario (base dos), pero ese número también podemos expresarlo en base diez, será: uno por uno más uno por dos más uno por cuatro más uno por ocho más uno por 16 más uno por 32 más uno por 64 más uno por 128 igual a: 255. Este es el número decimal más alto que podemos escribir en nuestros ocho interruptores; si estuviesen todos apagados el número que contendrían sería «cero». Si ha entendido lo explicado hasta aquí, no debe tener problemas en calcular a qué número decimal corresponde la configuración de interruptores 11Ø1ØØ1Ø; si no es capaz de resolverlo, vuelva a leer otra vez todo el artículo.



Cuestión de ejercitarse

Bien, como ya habrá adivinado, el número decimal correspondiente es 21Ø. Haga ejercicios con configuraciones diferentes hasta que domine el tema, y luego intente adivinar qué configuración de

interruptores se corresponde con los números decimales: 24, 255, 6Ø, 36, 66 y 129.

Si ha resuelto el ejercicio anterior, se encuentra usted en el camino de convertirse en un experto informático.

Supongamos, ahora, que tenemos un amigo en una fábrica de interruptores y nos regala todos los que queramos, así que cogemos una tabla muy grande y colocamos en ella 168 filas de ocho interruptores cada una, e total 1344 interruptores (tendremos que invitar un día a cenar a nuestro amigo). Ahora vamos a numerar las filas, a la primera fila la llamamos «cero», a la segunda la llamamos uno, y así sucesivamente hasta la última fila que se llamará 167. A estos números los llamamos «dirección», de modo que la dirección de la primera fila es «cero», la de la segunda es «uno», y así sucesivamente hasta la última, cuya dirección será 167. Al valor contenido en cada fila de interruptores lo llamamos «dato». Ahora tenemos lo que en informática se denomina una «memoria de 168 bytes con palabras de 8 bits». A un amigo nuestro le pedimos que introduzca en la dirección 37, el dato 21Ø; él busca la fila de interruptores que hemos llamado «37» y los coloca en la configuración 11Ø1ØØ1Ø.

Para definir gráficos es indispensable utilizar el comando POKE que trabaja directamente sobre la memoria

Cuando trabajamos con el ordenador, nuestro amigo se llama «intérprete de BASIC» y la forma de decirle que escriba el dato «21Ø» en la dirección «37» sería: POKE 37,21Ø. De hecho, nuestro ordenador dispone de una «tabla» de 65536 filas de ocho interruptores cada una, pero los interruptores de las 16384 primeras filas están fijos, y no podemos alterar su configuración.

Quizá se pregunte usted cómo es posible que más de medio millón de interruptores quepan en un espacio tan reducido. La respuesta inmediata es que son muy pequeños. De hecho, no son interruptores pero se comportan como si lo fueran, y el suponer que son interruptores facilita la compresión y no afecta para nada a nuestro problema.

128 64 32 16 8 4 2 1
USR "a" + Ø
USR "a" + 1
USR "a" + 2
USR "a" + 3
USR "a" + 4
USR "a" + 5
USR "a" + 6
USR "a" + 7

Fig. 1

Almacenar en memoria

No parece muy útil almacenar datos en la memoria si luego no podemos hacer nada con ellos. Sin embargo, todo lo que hace el ordenador depende de los datos almacenados en su memoria. Incluso las letras que salen en la pantalla están guardadas en la memoria del ordenador, ya que de lo contrario, éste no sabría escribirlas. Las letra y signos que forman el juego de caracteres del ordenador no podemos alterarlos, ya que están guardados en las 16384 primeras filas de interruptores. Pero nuestro ordenador tiene prevista la posibilidad de que nosotros le definamos una serie de caracteres para que trabaje con ellos como si se tratara de los suyos propios. Estos son los U.D.G. o gráficos definidos por el usuario, son 21 y la información para escribirlos está contenida en 168 filas de «interruptores» que sí podemos alterar.

En primer lugar necesitamos saber en qué filas de interruptores (o más propiamente, «posiciones de memoria») almacena el ordenador estos gráficos, y a continuación, deberemos saber qué datos hay que introducir en estas posiciones de memoria para que el ordenador nos imprima un determinado gráfico que nosotros hayamos creado. Vamos en primer lugar con el primero de los problemas.

El ordenador procura siempre guardar esta información en la parte de la memoria donde menos estorbe, y, por supuesto, donde no pueda ser borrada por el BASIC. El lugar adecuado son, por tanto, las 168 últimas posiciones de la memoria. Como se verá, son distintas según se trate de un modelo de 16 ó de 48 K.

Las direcciones de estas posiciones de memoria son: de la 326ØØ a la 32767 pra el modelo de 16K., y de la 65368 a la 65535 para el modelo de 48K. No se preocupe por tener que recordar estos números, ya que el ordenador los conoce, y nos los puede decir en cualquier momento si se lo sabemos preguntar. Cada carácter gráfico se almacena en ocho posiciones de memoria (por tanto, un carácter queda definido por los valores almacenados en 64 «interruptores»). La función USR «a» nos da la dirección de la primera de las posiciones de memoria correspondientes al gráfico definido por el usuario «a» (que es el primero de los U.D.G. y se obtiene con la «A» en modo gráfico). La función USR «b» nos dará la primera dirección del segundo gráfico, y así sucesivamente para los 21 gráficos posibles (hasta la «u»).

Datos a introducir

Ahora ya sabemos que para definir un gráfico tenemos que introducir ocho datos en memoria y las direcciones donde tenemos que introducirlos. Nos falta ser capaces de saber qué datos hay que introducir para generar un determinado gráfico.

Los gráficos que podemos generar han de ajustarse a un formato determinado, los haremos a base de puntos que estarán tan próximos que nos parecerán una figura. Cuantos más puntos utilicemos por unidad de superficie, más claramente se verá nuestra figura. La pantalla del Spectrum está compuesta por 45Ø56 puntos (en informática se les llama «pixels») dispuestos en un rectángulo de 255 por 176 puntos. Cada punto está unido a un «interruptor» en la parte de la memoria denominada «archivo de presentación visual» (en inglés «display file»), podemos imaginarnos que cada punto es una bombilla que está iluminada cuando su correspondiente interruptor está encendido.

La información para escribir los U.D.G. está contenida en 168 filas de «interruptores»

Cada segmento horizontal de ocho puntos está controlado por el dato almacenado en una determinada posición de memoria, y un conjunto de ocho segmentos colocados uno debajo de otro constituye un carácter que podrá ser una letra, un signo, o uno de nuestros gráficos U.D.G. (por cierto, U.D.G. son las iniciales de «User Defined Grafic», que en inglés significa «Grafico Definido por el Usuario»).

Cuando el ordenador imprime un carácter, lo que hace es trasladar los datos que figuran en los ocho bytes de memoria que definen ese carácter a los ocho bytes correspondientes del archivo de presentación visual.

Como vemos, cada carácter sea letra, signo o U.D.G., se compone de 64 puntos agrupados en un cuadrado de ocho por ocho; así que, si vamos a generar nuestro propio gráfico, deberemos empezar por coger una hoja de papel cuadriculado. Dibuje en ella un cuadrado como el de la figura 1, que abarque ocho por ocho cuadritos.

A continuación sombreamos los cuadritos necesarios para que aparezca el gráfico que queremos definir, supongamos que queremos que al pulsar la tecla «A» en modo «gráfico», aparezca el hombrecito de la figura 2, así que sombrearemos los cuadritos correspondientes. Recuerde que no puede sombrear un cuadrito a medias, ya que cada cuadrito corresponde a un interruptor, y cada interruptor sólo puede estar «encendido» o «apagado»; un cuadrito sombreado corresponde a un interruptor «encendido» y un cuadrito en blanco, a uno apagado.

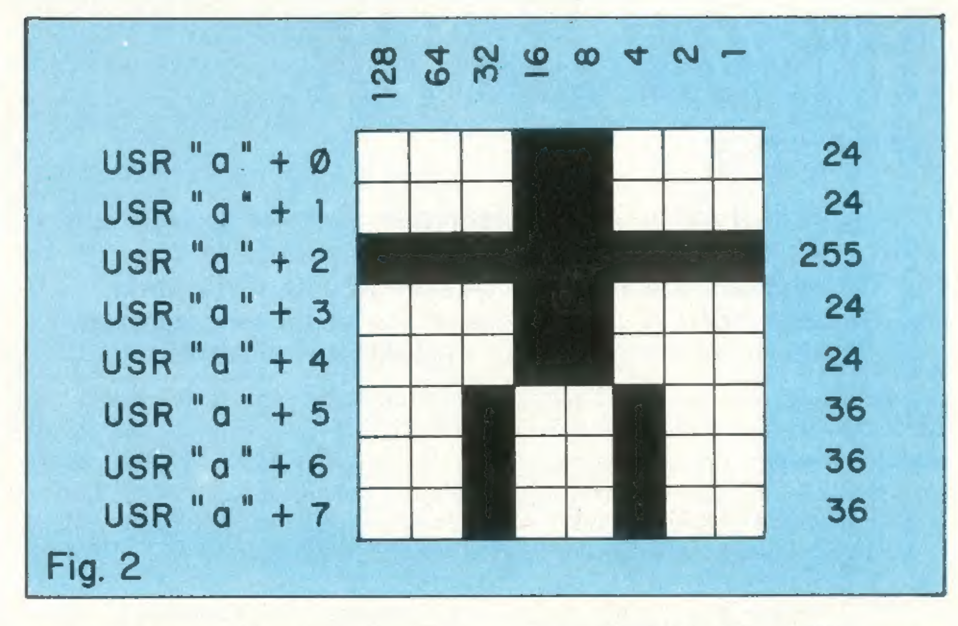


Fig. 2

Seguro que ya ha adivinado que cada fila horizontal de ocho cuadritos corresponde a una fila de ocho interruptores, es decir, a un byte. Si ponemos un «cero» donde hay un cuadro en blanco, y un «uno» donde hay un cuadro sombreado, seremos capaces de calcular cuál es el dato decimal que debe contener cada una de las ocho filas de ocho interruptores.

Dirección	Configuración binaria	Dato decimal
USR "a"+Ø	ØØØ11ØØØ	24
USR "a"+1	ØØØ11ØØØ	24
USR "a"+2	1 1 1 1 1 1 1 1	255
USR "a"+3	ØØØ11ØØØ	24
USR "a"+4	ØØØ11ØØØ	24
USR "a"+5	ØØ1ØØ1ØØ	36
USR "a"+6	ØØ1ØØ1ØØ	36
USR "a"+7	ØØ1ØØ1ØØ	36

En la primera columna «DIRECCION», tenemos la forma de obtener las direcciones donde almacenar los datos de la columna «DATO DECIMAL»; la columna central «CONFIGURACION BINARIA», nos da una idea de cómo quedarían colocados nuestros 64 interruptores.

Para introducir esta información en memoria podríamos hacer:

```
1Ø POKE USR "a"+Ø,24
2Ø POKE USR "a"+1,24
3Ø POKE USR "a"+2,255
4Ø POKE USR "a"+3,24
5Ø POKE USR "a"+4,24
6Ø POKE USR "a"+5,36
7Ø POKE USR "a"+6,36
8Ø POKE USR "a"+7,36
```

Un método mejor

Pero este método ocupa mucha memoria y es lento de teclear, así que vamos a pensar un método mejor:

```
1Ø FOR n=Ø TO 7
2Ø READ dato
3Ø POKE USR "a"+n,dato
4Ø NEXT n
5Ø DATA 24,24,255,24,24,36,36,36
6Ø PRINT AT 1Ø,1Ø;CHR$ 144
```

Las cuatro primeras líneas introducen nuestros datos en memoria, la quinta contiene la lista de datos y, la sexta, nos imprime en pantalla el carácter gráfico que acabamos de generar.

Si desea que el gráfico salga al pulsar la tecla "B" en modo gráfico, cambie la línea 3Ø por POKE USR "b"+n,dato y la línea 6Ø por PRINT AT 1Ø,1Ø;CHR$ 145, de esta forma puede almacenar 21 caracteres gráficos en las letras de la "A" a la "U" que corresponde a los caracteres con códigos comprendidos entre 144 y 164.

Cuando se conecta el ordenador, estas posiciones contienen una copia de los correspondientes caracteres en mayúsculas, así que no se extrañe si al pulsar la "F" en modo gráfico sin haber definido un gráfico para la "F", le sale precisamente una "F", ¿qué otra cosa podía esperar?

Confiamos en que después de leer este artículo esté en disposición de generar maravillosos gráficos para adornar sus programas. Si desea gráficos más grandes, puede hacerlo agrupando varios gráficos pequeños como seguramente habrá visto en más de un programa comercial, esta es la versión particular del Spectrum de los famosos «Sprites».

MISION LUNAR

Spectrum 48 K

En una situación de emergencia planetaria, en la cual los recursos minerales de nuestro mundo se han agotado, tenemos que emprender una arriesgada expedición al planeta vecino con el fin de establecer una base científica y de explotación industrial.

La tarea será ardua, porque el planeta de destino se encuentra rodeado de un cinturón de asteroides peligrosísimo de atravesar para cualquier astronave; es un mundo caliente e inhóspito, donde rocas aisladas, en las que tenemos que aterrizar, navegan sobre mares de lava fundida.

De cualquier forma, al principio del juego el ordenador de abordo nos informa de los peligros de la misión y de la forma de eludirlos.

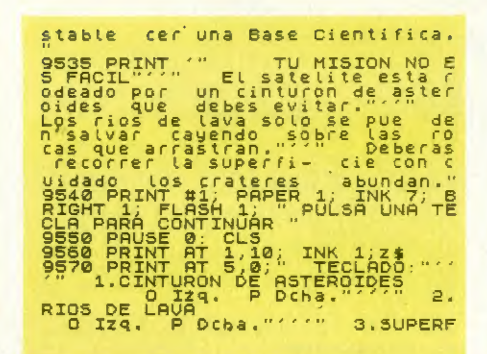
NOTAS GRAFICAS

A B C D E F G H I J

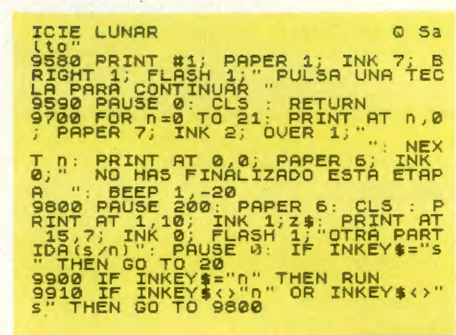
```
   1 PAPER 7: INK 7: BRIGHT 0: CLS
   2 GO SUB 9000
   3 LET z$="MISION LUNAR"
   4 PAPER 0: CLS : PRINT AT 10,3; INK 7;" M I S I O N   L U N A R "
   5 PRINT AT 11,0; INK 6;"                                "
  10 FOR w=0 TO 10: PRINT INK 3; AT w,16;" ";AT w-1,16;" ": PAUSE 3: NEXT w
  12 PAUSE 50: PRINT INK 2;AT 13,16;" "
  15 PRINT #0; PAPER 5; INK 0; FLASH 1; FLASH 0;"  DESEAS INSTRUCCIONES?  (s/n)  ": BEEP .1,10: BEEP .1,5: BEEP .1,0: BEEP .5,10
  18 PAUSE 0: IF INKEY$="s" THEN GO SUB 9500
  20 BORDER 3: INK 7: PAPER 0: BRIGHT 1: CLS
  30 LET xx=INT (RND*25)+5: LET yy=0
  32 FOR z=1 TO 150
  33 IF z>129 THEN PRINT AT 18,0;"": FOR n=1 TO 10: PRINT '''': POKE 23692,-1: GO TO 50
  35 LET a=INT (RND*30)+1
  40 PRINT AT 18,a; INK 6;"  ": PRINT AT 19,a; INK 6;"  "
  43 POKE 23692,-2
  48 PRINT ''''
  50 PRINT AT yy,xx;" "
  51 BEEP .01,(z/10)-5
  52 IF ATTR (yy+1,xx)=70 THEN GO TO 9700
  60 IF INKEY$="o" AND xx>1 THEN LET xx=xx-1
  70 IF INKEY$="p" AND xx<30 THEN LET xx=xx+1
 100 NEXT z
2000 CLS : FOR n=0 TO 21: PRINT AT n,xx;" ": PRINT AT n-1,xx;" ": BEEP .05,n: BEEP .05,n-10: NEXT n: CLS : PRINT AT 7,10;"ENHORABUENA": PRINT AT 12,3; PAPER 2; INK 7; FLASH 1;"AHORA ATRAVIESA LA LAVA": FOR n=1 TO 250: NEXT n
2030 BORDER 3: PAPER 5: INK 1: BRIGHT 1: CLS
2050 LET yy=0: LET xx=INT (RND*10)+20
2060 PRINT AT 20,0;"                                ": PRINT AT 21,0;"                                "
2070 LET z=0
2100 PRINT AT yy,xx; INK 0;" ": BEEP .01,-yy+10
2110 IF INKEY$="o" AND xx>0 THEN LET xx=xx-1: PRINT AT yy,xx+1;" "
2120 IF INKEY$="p" AND xx<31 THEN LET xx=xx+1: PRINT AT yy,xx-1;" "
2130: PRINT AT 19,z; INK 0;" ": PRINT AT 20,z; INK 0;" ": PRINT AT 19,z-1;" ": PRINT AT 20,z-1;" "
2140 LET z=z+1: IF z>30 THEN LET z=1: PRINT AT 19,30;" ": PRINT AT 20,30;" "
2150 IF yy>18 THEN GO TO 9700
2160 IF NOT yy=18 THEN GO TO 2170
2165 IF xx=z OR xx=z-1 OR xx=z+1 THEN GO TO 3000
2225 PRINT AT yy,xx;" ": LET yy=yy+.5
2300 GO TO 2100
3000 FOR n=z TO 31: PRINT AT 18,n; INK 0;" ": PRINT AT 18,n-1;" ": PRINT AT 19,n; INK 0;" ": PRINT AT 19,n-1;" ": PRINT AT 20,n; INK 0;" ": PRINT AT 20,n-1;" ": BEEP .03,n: BEEP .03,n-1: NEXT n
3005 PAPER 5: INK 4: BRIGHT 1: CLS : PRINT AT 10,2; PAPER 2; INK 7; FLASH 1;"BUSCA UN LUGAR PARA LA BASE": FOR n=1 TO 250: NEXT n
3010 RESTORE 3300
3015 LET a=20: LET p=0: LET xz=0
3020 LET a$="                                "
3025 LET b$="                                "
3030 LET c$="                                "
3040 LET d$="                                "
3050 FOR n=8 TO 21: PRINT AT n,0;"                                "
3060 NEXT n
3080 PRINT AT 21,0; PAPER 0;"                                "
3090 LET f$="                                "
3100 PRINT AT a,2; PAPER 4; INK 0;" "
3105 IF INKEY$="q" THEN LET p=1
3106 IF xz=135 THEN GO TO 7000
3107 LET xz=xz+1: LET e$=f$(xz TO 31+xz)
3110 PRINT AT 4,0;a$
3120 PRINT AT 5,0;b$
3130 PRINT AT 6,0;c$
3140 PRINT AT 7,0;d$
3145 PRINT AT 21,0; PAPER 4; INK 0;e$
3150 LET a$=a$(1 TO 32)+a$(1)
3160 LET b$=b$(1 TO 32)+b$(1)
3170 LET c$=c$(1 TO 32)+c$(1)
3180 LET d$=d$(1 TO 32)+d$(1)
3185 LET e$=e$(1 TO 32)+e$(1)
3190 LET a$=a$(2 TO )
3200 LET b$=b$(2 TO )
3210 LET c$=c$(2 TO )
3220 LET d$=d$(2 TO )
3225 LET e$=e$(2 TO )
3230 BEEP .01,-a+20
3235 IF a<>20 THEN GO TO 3245
3240 IF SCREEN$ (21,2)="_" THEN PRINT AT 20,1; PAPER 4;"  ": PRINT AT 21,2; PAPER 2; INK 7; FLASH 1;" ": PRINT AT 0,0; PAPER 6; INK 0;"    HAS FRACASADO EN TU MISION    ": BEEP 1,-20: PAUSE 100: GO TO 3005
3245 PRINT AT a,2; PAPER 4; INK 1;" "
3250 IF p=1 THEN READ a
3260 IF a=0 THEN LET p=0: LET a=20: RESTORE
3300 DATA 19,18,17,16,17,18,19,0
3500 GO TO 3100
4000 STOP
7000 REM felicitaciones
7010 PAPER 6: INK 0: BORDER 2: CLS
7020 FOR n=1 TO 21: PRINT AT n,0;" ": PRINT AT n-1,0;" ": NEXT n
7030 FOR n=0 TO 30: PRINT AT 21,n;"  ": NEXT n
7040 FOR n=20 TO 0 STEP -1: PRINT AT n,31;" ": PRINT AT n+1,31;" ": NEXT n
7050 FOR n=31 TO 0 STEP -1: PRINT AT 0,n;"  ": NEXT n
7060 PRINT AT 3,6;"HAS PASADO MUY BIEN ";AT 6,3;"EVITANDO LOS METEORITOS Y";AT 9,3;"ATERRIZANZDO CON SEGURIDAD";AT 15,7;"SOBREVIVIREIS !!!!!"
7070 FOR n=1 TO 30: BEEP .01,n: BEEP .01,n+3: BEEP .01,n+4: NEXT n
7080 PRINT AT 17,2; PAPER 2; INK 7; FLASH 1;"LA BASE LUNAR SERA UN EXITO!"
7500 GO TO 9800
9005 RESTORE 9020
9010 FOR n=1 TO 10: READ a$: FOR m=0 TO 7: READ a: POKE USR a$+m,a: NEXT m: NEXT n
9020 DATA "a",24,60,189,189,255,189,153,153
9030 DATA "b",195,135,31,63,127,127,255,255
9040 DATA "c",0,7,24,32,32,64,128,128
9050 DATA "d",0,224,24,4,4,2,1,1
9060 DATA "e",1,1,6,4,8,16,224,0
9070 DATA "f",128,128,96,32,16,8,7,0
9080 DATA "g",128,192,224,240,248,252,254,255
9090 DATA "h",1,3,7,15,31,63,127,255
9100 DATA "i",24,52,52,24,126,24,36,54
9110 DATA "j",24,24,60,60,126,126,255,255
9150 RETURN
9500 REM intrucciones
9510 BORDER 1: INK 0: PAPER 7: BRIGHT 1: CLS
9530 PRINT AT 1,10; INK 1;z$; INK 0;''"   Se han  agotado los recursos  minerales en tu planeta.      Debes  salir  en  expedicion  al planeta vecino para establece  cer una Base Cientifica."
```



J.R. BALLESTEROS



```
9535 PRINT '"       TU MISION NO ES FACIL"''"   El satelite esta rodeado por  un cinturon de asteroides  que  debes evitar."''"Los rios de lava solo se pueden salvar  cayendo  sobre las rocas que arrastran."''"   Deberas recorrer la superfi- cie con cuidado  los crateres   abundan."
9540 PRINT #1; PAPER 1; INK 7; BRIGHT 1; FLASH 1; " PULSA UNA TECLA PARA CONTINUAR "
9550 PAUSE 0: CLS
9560 PRINT AT 1,10; INK 1;z$
9570 PRINT AT 5,0;"  TECLADO:"''"   1.CINTURON DE ASTEROIDES         O Izq.   P Dcha."''"  2.RIOS DE LAVA    O Izq.   P Dcha."''"  3.SUPERFICIE LUNAR            Q Salto"
```



```
9580 PRINT #1; PAPER 1; INK 7; BRIGHT 1; FLASH 1;" PULSA UNA TECLA PARA CONTINUAR "
9590 PAUSE 0: CLS : RETURN
9700 FOR n=0 TO 21: PRINT AT n,0; PAPER 7; INK 2; OVER 1;"                                ": NEXT n: PRINT AT 0,0; PAPER 6; INK 0;"  NO HAS FINALIZADO ESTA ETAPA  ": BEEP 1,-20
9800 PAUSE 200: PAPER 6: CLS : PRINT AT 1,10; INK 1;z$: PRINT AT 15,7; INK 0; FLASH 1;"OTRA PARTIDA(s/n)": PAUSE 0: IF INKEY$="s" THEN GO TO 20
9900 IF INKEY$="n" THEN RUN
9910 IF INKEY$<>"n" OR INKEY$<>"s" THEN GO TO 9800
```

EL HORTELANO

Spectrum 48 K

Muchas han sido las misiones importantes y peligrosas que nos hemos visto obligados a realizar a lo largo de todas nuestras andanzas. En esta ocasión, la importancia de nuestro cometido es más humilde, pero no por ello menos arriesgada.



Por obra y gracia de nuestra imaginación, y de nuestro Spectrum, naturalmente, nos convertiremos en un laborioso hortelano que tiene que recoger el máximo número de hortalizas en el mínimo tiempo posible, tratando de evitar las múltiples minas que pueden estallar y acabar con nuestra vida. A pesar de la originalidad de este programa, poco más podemos decir de él, ya que las instrucciones «incluidas» en el mismo son lo suficientemente profusas y detalladas.

NOTAS GRAFICAS

A B C D E

[redacted]

```
   1 REM ZAMP  © Juan Fradera
   5 GO SUB 9000: GO SUB 9500: B
RIGHT 0
  10 LET l=0: LET c=0: LET t$="5
00": LET p=0: LET x=0: LET y=0:
LET v=3: LET pp=0
  90 BORDER 5: PAPER 4: INK 7: C
LS : GO SUB 8001: LET n=m+s+h
 100 INK 7: LET l=7: LET c=30: L
ET x=c: LET y=l
 110 REM Bucle de movimiento
 120 PRINT AT y,x;" ";AT l,c;"A"
;AT 1,10-LEN STR$ p; PAPER 7; IN
K 0;p;AT 1,18;"000"( TO 3-LEN t$
);t$: BEEP .0004,30: LET x=c: LE
T y=l
 150 LET l=l+(INKEY$="6")-(INKEY
$="7"): LET c=c+(INKEY$="8")-(IN
KEY$="5")
 155 LET p=p+(ATTR (l,c)=34): LE
T t$=STR$ (-1+(20 AND ATTR (l,c)
=35)+VAL t$): GO TO 200+(1000 AN
D ATTR (l,c)=38)+(1250 AND t$="-
1")+(2000 AND ATTR (l,c)=35)+(30
00 AND p=250-n)+(4100 AND ATTR (
l,c)=37)+(4800 AND ATTR (l,c)=32
)
 200 GO TO 120
1200 REM Pisar una mina
1250 PRINT AT y,x;" ";AT l,c;"A"
: IF INT (RND*3)=1 THEN PRINT AT
 2,7; FLASH 1; PAPER 5; INK 0;"L
a mina no estallo": FOR g=1 TO 1
0: FOR f=1 TO 10: BEEP .01,f: NE
XT f: NEXT g: BORDER 5: PRINT AT
 2,7; PAPER 5;"
": GO TO 120
1300 PRINT AT 2,1; FLASH 1; PAPE
R 5; INK 0;"B Lo siento, la mina
 estallo B": FOR f=1 TO 20: PRIN
T AT l,c; INK (7 AND f/2=INT (f/
2));"A": BEEP .005,20: BEEP .004
,15: BEEP .003,30: BEEP .002,25:
 NEXT f: PRINT AT 2,1; PAPER 5;"
                              "
1350 REM Muerte
1400 LET v=v-1: PRINT AT 1,31-v;
 PAPER 7;" ": IF v<>0 THEN GO TO
 120
1430 GO TO 1500
1450 PRINT AT 2,5; FLASH 1; PAPE
R 5; INK 0;"A Se acabo el tiempo
 A": BEEP 4,20
1500 LET pp=pp+p: PRINT AT 2,0;
PAPER 5; INK 0;"          T.S
CORE:              ";AT 2,18;pp
*10;AT 11,12; PAPER 0; INK 7;"GA
ME OVER";AT 12,14;"(s/n)"
1600 IF INKEY$="s" OR INKEY$="S"
 THEN RUN
1650 IF INKEY$="n" OR INKEY$="N"
 THEN STOP
1700 GO TO 1600
2200 REM Coger una Super Fruta
2300 BEEP .005,20: BEEP .005,10:
 GO TO 120
2700 GO TO 1200: REM Enlace de
final de tiempo y pisar mina
3200 REM Fin de la recoleccion
3300 IF l<>7 OR c<>30 THEN GO TO
 120
3400 IF v<>3 THEN LET v=v+1
3450 PRINT AT y,x;" ";AT l,c;"A"
; FLASH 1; PAPER 5; INK 0;AT 2,0
;"C Muy  bien, lo has conseguido
 C": FOR f=1 TO 15: FOR g=10 TO
14: BEEP .05,g: NEXT g: BORDER I
NT (f/2): NEXT f: BORDER 5
3475 LET pp=pp+p: LET p=0: CLS :
 INK 7: LET t$="500": GO SUB 800
1
3500 LET n=h+m+s: GO TO 100
4200 GO TO 1200: REM Enlace del
fin de frutas y pisar mina
4250 REM Saltar el charco
4300 LET cc=-(INKEY$="5")+(INKEY
$="8"): LET ll=-(INKEY$="7")+(IN
KEY$="6"): LET c1=c+2*cc: LET l1
=l+2*ll: IF cc=0 AND ll=0 THEN G
O TO 5100
4350 IF ATTR (l1,c1)=32 AND l1=l
+ll AND c1=c+cc THEN LET n=n+INT
 (p/5): LET p=p-INT (p/5): PRINT
 AT 2,4; PAPER 5; INK 0;"E Caist
es en el charco E": FOR f=1 TO 4
: BEEP .005,20: BEEP .005,0: BEE
P .005,10: NEXT f: PRINT AT 2,4;
 PAPER 5;"
 ": GO TO 120
4356 IF ATTR (l1,c1)=32 THEN PRI
NT AT l,c;(" " AND b$="3"): LET
l=l+ll: LET c=c+cc: BEEP .005,30
: BEEP .005,10: GO TO 155
4400 IF ATTR (l1,c1)<>32 AND l1>
2 AND l1<21 AND c1>0 AND c1<32 T
HEN PRINT AT l,c;(" " AND b$="3"
): LET c=c1: LET l=l1: BEEP .005
,30: BEEP .005,10: GO TO 155
4500 LET c1=c+cc: LET l1=l+ll: G
O TO 4350
4700 GO TO 3200: REM Enlaca de
final del tiempo y final de fru-
tas
5000 REM Pisar la balla
5100 LET l=y: LET c=x: GO TO 120
5300 GO TO 2200: REM Enlace del
fin de frutas y pisar s.fruta
6350 GO TO 1450: REM Enlace del
fin de tiempo y pisar charco,y
fin de tiempo y pisar valla,y de
final de tiempo, final frutas y
pisar mina
6450 GO TO 2200: REM Enlace del
fin de tiempo, fin de frutas y
pisar s.fruta
7500 GO TO 4300: REM Enlace  del
fin de frutas y pisar charco
8000 GO TO 5100: REM Enlace del
fin de frutas y la valla.
   : REM ©, marcador y recinto
8001 LET a$="####################
############": PRINT AT 0,0; FL
ASH 1; INK 0; PAPER 5;"       ZAM
P  © Juan Fradera            "
8002 PRINT PAPER 7; INK 0;"SCORE
 00000  TIME 500  LIVES AAA"; PA
PER 5;"
 "
8003 PRINT AT 1,28; PAPER 7;"   "
( TO 4-v)
8005 PRINT INK 0;AT 3,0;a$;AT 21
,0;a$: FOR f=3 TO 21: PRINT INK
0;AT f,0;"#";AT f,31;"#": NEXT f
8009 REM Frutas
8010 FOR c=4 TO 20: FOR f=1 TO 2
9 STEP 2: PRINT INK 2;AT c,f;"C"
: NEXT f: NEXT c
8025 PRINT INK 7;AT 6,29;"  ";AT
 4,29;"  ";AT 5,28;"   ";AT 7,29
;"  "; INK 0;AT 4,28;"#";AT 6,30
;"#";AT 6,28;"#";AT 7,28;"#";AT
8,28;"###"
8040 REM Minas
8050 LET m=INT (5+RND*20): FOR x
=1 TO m
8070 LET l=INT (4+RND*19): LET c
=INT (RND*32): IF ATTR (l,c)=34
THEN PRINT INK 6;AT l,c;"B": NEX
T x: GO TO 8300
8080 GO TO 8070
8410 REM Super Frutas
8425 LET s=INT (3+RND*10): FOR x
=1 TO s
8450 LET l=INT (4+RND*19): LET c
=INT (RND*32): IF ATTR (l,c)=34
THEN PRINT INK 3;AT l,c;"D": NEX
T x: GO TO 8500
8460 GO TO 8450
8500 REM Charco
8510 GO TO 8570
8550 GO TO 1450: REM Enlace del
fin de tiempo, fin de frutas y
pisar charco
8570 LET h=0: IF b$="1" THEN RET
URN
8600 LET h=INT (5+RND*20): FOR x
=1 TO h
8650 LET l=INT (4+RND*19): LET c
=INT (RND*32): IF ATTR (l,c)=34
THEN PRINT INK 5;AT l,c;"E": NEX
T x: RETURN
8700 GO TO 8650
9000 REM Def. de caracteres
9100 FOR f=0 TO 7: READ a: POKE
USR "a"+f,a: NEXT f: REM Hombre
9150 FOR f=0 TO 7: READ a: POKE
USR "b"+f,a: NEXT f: REM Mina
9200 FOR f=0 TO 7: READ a: POKE
USR "c"+f,a: NEXT f: REM Fruta
9225 GO TO 9275
9250 GO TO 1450: REM Enlace del
fin de tiempo , fin de frutas y
pisar valla
9275 FOR f=0 TO 7: READ a: POKE
USR "d"+f,a: NEXT f: REM S.Fruta
9300 FOR f=0 TO 7: READ a: POKE
USR "e"+f,a: NEXT f: REM Charco
9310 FOR f=0 TO 7: READ a: POKE
USR "k"+f,a: NEXT f: REM EKe
9400 DATA 16,122,16,122,186,63,4
2,108,0,60,66,90,90,66,60,0,0,0,
2,12,60,56,24,0,0,16,16,214,124,
56,56,0,0,102,127,62,62,124,124,
24,120,0,120,68,68,68,68,0
9450 RETURN
9500 REM Presentecion e instruc-
          ciones
9505 PAPER 4: BRIGHT 1: CLS
9510 FOR l=7 TO 0 STEP -1
9600 BORDER 7-l: INK l: PRINT AT
 6,0;"BBBBBBBBBBBBBBBBBBBBBBBBBB
BBBBBB
       BBBBBBBBBBBBBBBBBBBBBBBBB
BBBBBB"
9605 PRINT INK 9;"        © Juan
 Fradera          ";AT 21,0;"Qui
eres instrucciones (s/n)?"
9610 BEEP .01,l-7: IF INKEY$="s"
 OR INKEY$="S" THEN BEEP 1,20: G
O TO 9700
9620 IF INKEY$="n" OR INKEY$="N"
 THEN : BORDER 4: INK 0: PAPER 4
: CLS : GO TO 9870
9650 NEXT l
9670 GO TO 9510
9700 PAPER 4: INK 0: BORDER 4: C
LS : PRINT AT 0,0;"         INST
RUCCIONES
                    Se trata de u
n juego de habili- dad y rapidez
.                 Tu eres un ca
mpesino que tienes que recoger t
u cosecha antes de la venida del
 mal tiempo.       "
9710 PRINT "Pero ademas de no po
der pasarte del tiempo  hay otro
 problema..,unos gamberros te hi
n puesto en el campo unas minas
que pueden  explotar, por lo tan
to esquiva- las o saltalas,(ya t
e explicare como).
 "
9720 PRINT "Por cada fruto que c
ojas obten- dras 10 puntos. Tien
es 3 vidas, y por cada mina que
pises y ex- plote te quedara una
 menos.     Para empezar a recog
er la cose- cha tienes que salir
 de tu caba-ña, y una vez termin
ado el tra- bajo tendras de volv
er a ella."
9750 IF INKEY$="" THEN GO TO 975
0
9760 BEEP 1,20: CLS : PRINT "En
el campo podras encontrar   uno
s vegetales que si los coges el
mal tiempo llegara mas tarde.Has
 de tener en cuenta que no   pue
des salir del campo, ni sal- tan
do las vallas.                 "
9770 PRINT "Los puntos que vayas
 obteniendo en cada pantalla se
te iran su- mando, y el total ob
tenido se temostrara al finaliza
r el juego. En el campo, (segun
la opcion   que cojas), hay unos
 charcos quelos puedes saltar en
 la direc-  cion que quieras, e
incluso, conpractica y rapidez d
e dedos     podras saltar con ef
ecto.     "
9780 PRINT "Si intentas saltar v
arios char- cos seguidos tendras
 que coger  fuerza manteniendo p
ulsada la   tecla en la direccio
n del salto."
9790 IF INKEY$="" THEN GO TO 979
0
9800 BEEP 1,20: CLS : PRINT "Sol
o se pueden saltar charcos, ysi
es posible ,(por no encon-  tra
rte delante de la valla), se sal
tara algo mas alla de este.  Si
saltas un charco que se en-  cue
ntre inmediatamente delante  de
la valla, caeras al agua y   per
deras una quinta parte de lospun
tos acumulados en esa panta- lla
.                              "
9810 PRINT "Estas son las teclas
 de control:              5>Izquierd
a                         8>Derecha
                          7>Arriba
                          6>Abajo
          "
9820 PRINT "Y esto lo que veras
en pantalla:": PRINT ;TAB 10; IN
K 7;"A >Tu, el campesino";TAB 10
; INK 6;"B >Las minas";TAB 10; I
NK 2;"C >La cosecha";TAB 10; INK
 2;"D >Los vegetales";TAB 10; IN
K 5;"E >Los charcos";TAB 10; INK
 0;"# >Las vallas"
9850 IF INKEY$="" THEN GO TO 985
0
9870 BEEP 1,20: CLS : PRINT "Aho
ra elije la opcion que deseesy .
... !Manos a la azada!"'"Opcion
es:"
9890 PRINT '"1) Sin charcos."'"2
) Normal."'"3) Una vez saltado u
n charco        este se seca."
9900 LET b$=INKEY$: IF b$="" THE
N GO TO 9900
9905 IF b$<>"1" AND b$<>"2" AND
b$<>"3" THEN BEEP 1,5: GO TO 990
0
9910 PRINT '"La opcion deseada e
s la ";b$: FOR l=1 TO 75: BEEP .
01,60-l/2: NEXT l: BEEP .1,20: B
EEP .1,5: RETURN
9999 SAVE "Zamp" LINE 1
```

[redacted]

EL DEFENSOR

Spectrum 16 K

Tenemos, en esta ocasión, un programa «arriesgado» y trascendental para la supervivencia de nuestro planeta. Con esta responsabilidad sobre nuestras espaldas, deberemos tratar de defender la tierra de los ataques exteriores.



J.R. BALLESTEROS

Una vez más, la amenaza ALIEN se cierne sobre nuestro planeta. Aunque se trate de extraterrestres saltarines, no por ello son menos peligrosos; si atraviesan nuestras defensas, destruirán la Tierra. Por tanto, nuestra misión consistirá en aumentar la barrera defensiva más rápido de lo que los alienígenas la aniquilan; así que mucha velocidad en sus movimientos y ¡cuidado!, el «malo» recibirá refuerzos en el momento más inesperado.

NOTAS GRAFICAS

A B C D E F G H I J

[redacted]

```
  50 INK 7: PAPER 0: BORDER 0: C
LS
  60 GO SUB 9000
  70 GO SUB 8000
  80 PRINT AT 0,0; PAPER 2; INK
6;" PUNTOS          RECORD  0
 "
  90 LET hi=0
 100 GO SUB 4000
 110 BEEP 1,30: PRINT AT 2,10; F
LASH 1;"ALERTA ALIEN": PAUSE 100
: PRINT AT 2,10;"            ":
PAUSE 50
 120 LET sc=0
 130 LET p=15: LET q=1: LET f=0:
 LET g=1
 140 LET v=1: LET w=1: LET h=1:
LET i=1
 150 LET x=15: LET y=20
 160 LET a=x: LET b=y
 170 LET y=y-(INKEY$="q" AND y>3
)+(INKEY$="a" AND y<20)
 180 LET x=x-(INKEY$="o" AND x>0
)+(INKEY$="p" AND x<31)
 190 IF ATTR (y,x)=5 THEN BEEP .
01,40
 200 PRINT AT b,a;" ";AT y,x; IN
K 7;"A"
 210 PRINT AT q,p;" ";AT q,p-1;"
 "
 220 IF INKEY$="m" THEN PRINT AT
 y-1,x; INK 5;"C"
 240 LET p=p+v: LET q=q+w
 250 IF p=31 OR p=0 THEN LET v=-
v: BEEP .01,15
 260 IF q=1 THEN LET w=-w: BEEP
.01,15
 270 IF ATTR (q,p)=5 THEN BEEP .
01,50: GO SUB 3000
 280 IF ATTR (q,p)=4 THEN GO TO
7000
 290 PRINT AT q,p; INK 6;"B"
 300 IF sc<1500 THEN GO TO 380
 310 PRINT AT g,f;" ";AT g,f-1;"
 "
 320 LET f=f+h: LET g=g+i
 330 IF f=31 OR f=0 THEN LET h=-
h: BEEP .01,15
 340 IF g=1 THEN LET i=-i: BEEP
.01,15
 350 IF ATTR (g,f)=5 THEN BEEP .
01,50: GO SUB 3500
 360 IF ATTR (g,f)=4 THEN GO TO
7000
 370 PRINT AT g,f; INK 6;"B"
 380 LET sc=sc+1: PRINT AT 0,8;
PAPER 2; INK 6;sc
 390 GO TO 160
 400 IF sc>=hi THEN LET hi=sc: P
RINT AT 0,25; PAPER 2; INK 6;hi:
 PRINT AT 0,8; PAPER 2; INK 6;"0
 "
 410 FOR z=1 TO 20
 420 PRINT AT z,0; INK 0;"[redacted]"
 430 NEXT z
 440 GO TO 100
3000 LET v=v: LET w=-w
3010 IF q=21 OR q=0 THEN LET w=-
w
3020 RETURN
3500 LET h=h: LET i=i
3510 IF g=21 OR g=1 THEN LET i=-
i
3520 RETURN
4000 FOR s=1 TO 30: FOR t=7 TO 1
3 STEP 2
4010 PRINT AT t,s; INK 5;"C": BE
EP .01,s
4030 NEXT t: NEXT s
4040 FOR j=0 TO 31
4050 PRINT AT 21,j; INK 4;"D": B
EEP .01,2*j
4060 NEXT j
4070 PRINT AT 2,10; INK 0;"[redacted]"
4080 RETURN
7000 IF ATTR (q,p)=4 THEN PRINT
; INK 6; BRIGHT 1;AT q-1,p-1;"FH
J";AT q-2,p-1;" G ": BEEP .5,1:
BEEP .5,1: BEEP 2,-2: PRINT AT q
-1,p-1; INK 0;"[redacted]";AT q-2,p-1;
INK 0;"[redacted]"
7010 IF ATTR (g,f)=4 THEN PRINT
; INK 6; BRIGHT 1;AT g-1,f-1;"FH
J";AT g-2,f-1;" G ": BEEP .5,1:
BEEP .5,1: BEEP 2,-2: PRINT AT g
-1,f-1; INK 0;"[redacted]";AT g-2,f-1;
INK 0;"[redacted]"
7020 PRINT AT 21,0; PAPER 4; INK
 7; BRIGHT 1; FLASH 1;"▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
7030 PRINT AT 3,11; PAPER 1; INK
 7; FLASH 1;"GAME OVER"
7040 FOR b=60 TO -20 STEP -2
7050 BEEP .01,b
7060 NEXT b
7070 INPUT "     Otro juego? (s/n
)"; LINE a$
7080 IF a$="s" OR a$="S" THEN GO
 TO 400
7090 IF a$="n" OR a$="N" THEN CL
S : GO TO 7110
7100 IF a$<>"s" OR a$<>"S" OR a$
<>"n" OR a$<>"N" THEN GO TO 7070
7110 FOR x=0 TO 15: PRINT '' INK
 RND*6;TAB RND*15;"OK,!!!!": POK
E 23692,255: NEXT x: CLS
8000 FOR b=1 TO 7: BEEP .2,30: P
RINT AT b,8; INK 0; PAPER b;" D
E F E N D E R ": NEXT b
8003 PRINT AT 10,12; INK 5; BRIG
HT 1;"CONTROLES"
8005 PRINT AT 12,0; PAPER 2; INK
 6;"SUBIR BAJAR  DECHA IZQDA  VA
LLAR
  "; INK 5; PAPER 3;"  Q      A
    O      P      M   "
8007 FOR c=1 TO 14: PRINT ; PAPE
R 6; INK 3;AT 20,c;" PULSA UNA T
ECLA>": NEXT c: PAUSE 0
8008 CLS
8009 PRINT AT 2,0;''"  Quien est
a destrozando la pro-teccion de
la tierra?"; BRIGHT 1; INK 3;"
 "
8010 PRINT ;'"  Cuanto podra res
istir el pobreJoe defendiendola
contra el infatigable alien?."
8020 PRINT ;'"  Trata de retrasa
r su  invasionaumentando la barr
era de protec-cion."
8030 PRINT ;'"  Podras evitar el
 acoso del se-gundo alien???"
8060 FOR c=0 TO 31: PRINT BRIGHT
 0;AT 19,c; PAPER 1; INK 7;"
 PULSA UNA TECLA PARA JUGAR>":
NEXT c
8070 PAUSE 0: CLS
8080 RETURN
9000 FOR n=1 TO 10
9001 FOR d=1 TO 5: PRINT FLASH 1
;AT 7,8;" D E F E N D E R ": BEE
P .05,15: NEXT d: FLASH 0
9010 READ a$
9020 FOR p=0 TO 7
9030 READ q: POKE USR a$+p,q
9040 NEXT p
9050 NEXT n
9060 DATA "A",3,27,27,15,63,27,1
20,64,"B",66,60,90,126,126,90,66
,36,"C",28,34,73,89,77,73,34,28,
"D",241,159,241,159,241,159,241,
159,"E",0,0,2,2,9,37,16,8,"F",10
0,2,225,24,70,0,252,0,"G",16,146
,84,84,85,17,130,130,"H",84,84,1
7,186,124,124,254,16,"I",0,0,128
,128,32,72,16,32,"J",76,128,14,4
,8,196,0,126,0
9700 RETURN
```

[redacted]

MUGSY

La compañía Melbourne House, que ya ha demostrado en numerosas ocasiones su eficacia programando aventuras, (recordemos si no el Hobit, o más recientemente Srlock Holmes), ha creado en esta ocasión un videocomic en forma de aventura, «Mugsy», un juego con excelentes gráficos.



Melbourne/Ibeson
48 K
Tipo de juego: Videocomic
PVP: 1.900 ptas.

Nuestro papel en el juego es el de representar a Mugsy, el padrino de una pandilla de pillos, con el fin de convertirnos en el más duro y poderoso gánster de la ciudad. El objetivo es dirigir una organización intentando multiplicar nuestro dinero, que aumentará o disminuirá según las decisiones que tomemos, las cuales influyen directamente también en nuestro grado de poder.
La protección de nuestros clientes será obra de las decisiones a tomar cada año. Nuestra mano derecha nos irá informando puntualmente del número total de los que necesitan protección y nosotros elegiremos a cuántos vamos a proteger. Como es lógico, esta decisión tendrá repercusiones económicas, unas veces positivas y otras negativas, según se desarrollen los acontecimientos.
Otra de las decisiones que tenemos que tomar, es el dinero que vamos a dedicar a comprar armas. Hay que tener mucho cuidado con este aspecto porque si destinamos mucho dinero a armas, nos podemos quedar sin dinero, pero si en cambio empleamos poco, corremos el riesgo de perder el control.
Los sobornos a la gente del sindicato van a ser igualmente decisivos a la hora de calibrar nuestro poder, mientras que, por otro lado, el dinero que destinemos a golpes va a hacer que aumente o disminuya el capital disponible.
Cuando tengamos demasiado éxito, otros ganster intentarán aprovecharse de nuestra situación y si fracasamos, aparecerá una secuencia del tipo arcade en la que tendremos que luchar contra nuestro enemigo para intentar salvarnos.
Los gráficos del juego son muy buenos, están diseñados como si fuera un comic. La mecánica en este sentido, se basa en la relación pregunta-respuesta. Normalmente la figura que hay junto a nosotros es la que más habla, y nos informa de la situación de la organización. Nuestro personaje, sin embargo, casi siempre aparece de espaldas, dando una sensación de este modo aún más real, como si fuéramos nosotros los que ocupáramos ese lugar.



Cuando es nuestro turno de hablar, aparece un cursor intermitente en el lugar donde tenemos que responder.
El juego se desarrolla en varias pantallas diferentes, todas ellas en ambiente gatsby, simulando el Chicago de los años veinte.
Al inicio del programa podemos escuchar una melodía muy agradable, sobre todo por lo bien que está hecha.





Es un buen juego en el que la ambientación y los gráficos cumplen a la perfección con el contenido.
El programa está en inglés coloquial lo que puede dificultar un poco su comprensión. De todas formas, en las instrucciones se traducen las fases más importantes del juego.

BLUE THUNDER

Richard Wilcox/ERBE
48 K.
Tipo de juego: Arcade.
PVP: 1.500 ptas.

Es un juego bastante interesante. Aunque al principio no parece gran cosa, sin embargo, a medida que vamos avanzando va resultando más completo cada vez.
Basado en la mecánica típica de los juegos de arcade, donde los enemigos suelen ser innumerables y los peligros contínuos, el juego resulta vibrante de emoción en las diferentes fases por las que vamos atravesando.
Hay cinco etapas a las cuales podemos acceder según deseemos pulsando previamente la que hayamos seleccionado, en cada una de las cuales tendremos que completar una misión, que nos resultará cada vez más difícil, según la fase sea más alta.
Nada más comenzar el juego, aparece nuestro helicóptero sobre la plataforma de un barco y tenemos que hacerle despegar y dirigirle hasta el centro neurálgico de las posiciones del enemigo. Para ello, será necesario haber atravesado con anterioridad todas sus líneas evitando el fuego incesante de sus baterías de costa y de los rayos destructores que nos lanzan durante todo el trayecto, hasta llegar a una zona donde nos encontraremos con un campo de fuerza que no podemos atravesar si antes no hemos destruido sus sistemas de protección.
Cuando lo hagamos, una plataforma surgirá del mar y habrá que penetrar en ella con el fin de desactivar los sistemas de defensa. Una vez conseguido, proseguirá nuestra misión que está llena de sorpresas de todo tipo.
El juego tiene un estilo parecido al Harrier Attack, pero en esta ocasión, con muchas más ideas llenas de originalidad que, junto a los buenos gráficos que tiene, le hacen un juego muy interesante para todo tipo de usuarios. Como decíamos al principio, cuando comienza parece mucho más simple de lo que luego en realidad resulta. A medida que nos vamos adentrando en el juego, los gráficos son mejores y el nivel de dificultad aumenta.
El único inconveniente que le hemos visto es el movimiento, que resulta un tanto lioso al principio hasta que logramos hacernos con los mandos del helicóptero, sobre todo para girar y disparar, ya que ambas operaciones se realizan con la misma tecla, lo que dificulta un poco más el juego.
El helicóptero esta muy bien conseguido y, cuando vuela en posición lateral, consigue un efecto muy vistoso que le dota de más realismo.
En definitiva, un juego de arcade que se sale un poco de la tónica general y con el que se puede pasar un rato muy entretenido.



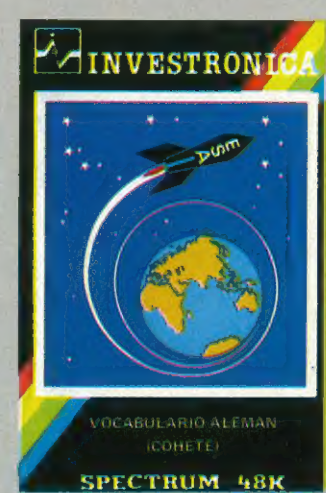




VOCABULARIO ALEMAN (COHETE)

Investrónica.
48 K.
Tipo: Educativo.
PVP: 1.900 ptas.

Se trata de un programa muy parecido al que comentamos con anterioridad en esta misma sección, con la diferencia de que, en esta ocasión, el idioma que trata es el alemán y el objetivo es poner un cohete en órbita, lo cual sólo se conseguirá si antes resolvemos la palabra clave que podrá estar en Alemán, Castellano o indistintamente, a elección del ordenador en uno de los dos idiomas.
Este programa pertenece al CAI (Computer Aided Instruction) que forma parte de un plan de enseñanza asistida por ordenador, que es, en realidad, lo que significan sus siglas. Está basado en un método que pretende que sea el propio alumno el que vaya descubriendo los conocimientos por sí solo de una forma sencilla, como si de un simple juego se tratase. Es un programa compatible con cualquier método de aprendizaje de idiomas y permite, para los que ya conozcan algo de éste, una puesta al día en su vocabulario.



El juego está concebido en forma de adivinación, en la pantalla van apareciendo las palabras que tienen que ser traducidas. Se conoce la longitud de éstas, marcada por el número correspondiente de guiones, donde posteriormente se insertarán las letras que vayamos tecleando.
El máximo de fallos permitidos es de diez letras incorrectas, teniendo en cuenta que nosotros podemos delimitar éstas si lo deseamos, para dificultar aún más el juego y dotarle de un mayor interés a medida que avancemos en nuestros conocimientos.
Al principio del juego hay un pequeño menú que nos indica las teclas que tenemos que pulsar para poder utilizar algunas letras que se usan en ambos idiomas y que no vienen en nuestro teclado, como por ejemplo la «ñ».
El vocabulario de que disponemos es de 1.800 palabras que han sido extraídas de la primera acepción de un prestigioso diccionario alemán. Es aconsejable sin embargo, si no se tiene mucho dominio reducirlo al principio a un máximo de 400 palabras. Se puede de esta forma, programar un curso por niveles e ir incrementando éstos a medida que avanzamos en nuestros estudios.
Muy recomendado para estudiantes de idiomas o para aquellos que se inicien en esta materia.

RIDER

Virgin/Compulogical.
48 K.
Tipo de juego: Arcade.
PVP: 1.550 ptas.

Es un juego bastante simple que consiste en ir recorriendo un camino sembrado de minas, evitando éstas para tratar de

INICIO → ASIGNACION DEL COLOR DE PANTALLA → DIBUJO DEL CONTORNO → VISUALIZACION DE LOS ROTULOS → ENTRADA Y VISUALIZACION DE DATOS → FIN

DATOS
NOMBRE
APELLIDOS
CALLE
NUMERO
PISO
PUERTA
TELEFONO
POBLACION

PROGRAMA "FICHA"

Programa "Ficha".

Este programa simula una ficha con diversos *campos* (nombre, apellidos, etc...) que el usuario debe rellenar. Este programa puede formar parte de otro mayor que almacene, por ejemplo, los datos de los empleados de una empresa, los datos de los clientes o los suministradores.

Al editar el programa, debe poner atención en las líneas 5ϕ a 11ϕ, ya que van incluidos algunos de los gráficos predefinidos que incorpora el Spectrum. Debe pasar a modo G (gráficos).

Las líneas 8ϕ a 1ϕϕ forman lo que en programación se llama *bucle*, por tanto la instrucción 9ϕ se repetirá 2ϕ veces pero con diversos valores de la variable «n», ya que el índice del bucle está comprendido entre los márgenes 1 y 2ϕ. Los bucles «FOR»...«NEXT» serán estudiados con mayor detalle en otro capítulo.

La sentencia «FLASH 1» antepuesta al símbolo « < » (menor que) hace que este parpadee en la pantalla para llamar la atención sobre el dato a introducir.

Y por último la sentencia «PAUSE 2ϕϕ» temporiza, aproximadamente, la ejecución del programa durante cuatro segundos.

La estructura del programa es la siguiente:

1ϕ	: Comentario con el nombre del programa.
2ϕ	: Asignación de los colores de la pantalla, azul, para el borde y papel, y amarillo para los caracteres.
5ϕ — 11ϕ	: Dibujo del contorno con los gráficos predefinidos.
13ϕ — 21ϕ	: Visualización de los campos de la ficha.
22ϕ	: Temporización.
24ϕ — 5ϕϕ	: Entrada de los datos y visualización de estos en los campos correspondientes.
51ϕ	: Utilización del canal de comunicación ϕ para visualizar el informe «Fin de edición».
52ϕ	: Temporización.

Programas de repaso

Como colofón al estudio de las sentencias básicas de programación:

```
REM
LET
PRINT
INPUT
```

se analizan en este capítulo cinco programas realizados con este tipo de instrucciones.

Estos programas, numerados del uno al cinco, son los siguientes:

1. GRANJA
2. ECUACION
3. INTERES
4. GRADOS
5. FICHA

Programa «GRANJA»

Para almacenarlo en cinta una vez editado hágalo, por ejemplo, de la forma:

```
SAVE "granja"
```

Este programa es bastante sencillo, ya que calcula el número total de animales que hay en una granja, a partir de los datos que le son introducidos por teclado; también calcula el número total de patas.

La sentencia 1ϕ es la presentación del programa. En la 2ϕ hay cuatro sentencias, esto es posible ya que se utiliza el signo separador «:»; éstas no se han explicado todavía, pero vamos a ver unas pequeñas nociones sobre su funcionamiento. En conjunto realizan la tarea de colorear la pantalla. La sentencia «BORDER 1» asigna el color *azul* al borde de la pantalla, «PAPER 1» asigna el color azul al fondo, e «INK 7» tiene la misión de que el color de los caracteres sea

TRUE VIDEO

COLOR DEL CARACTER (INK)

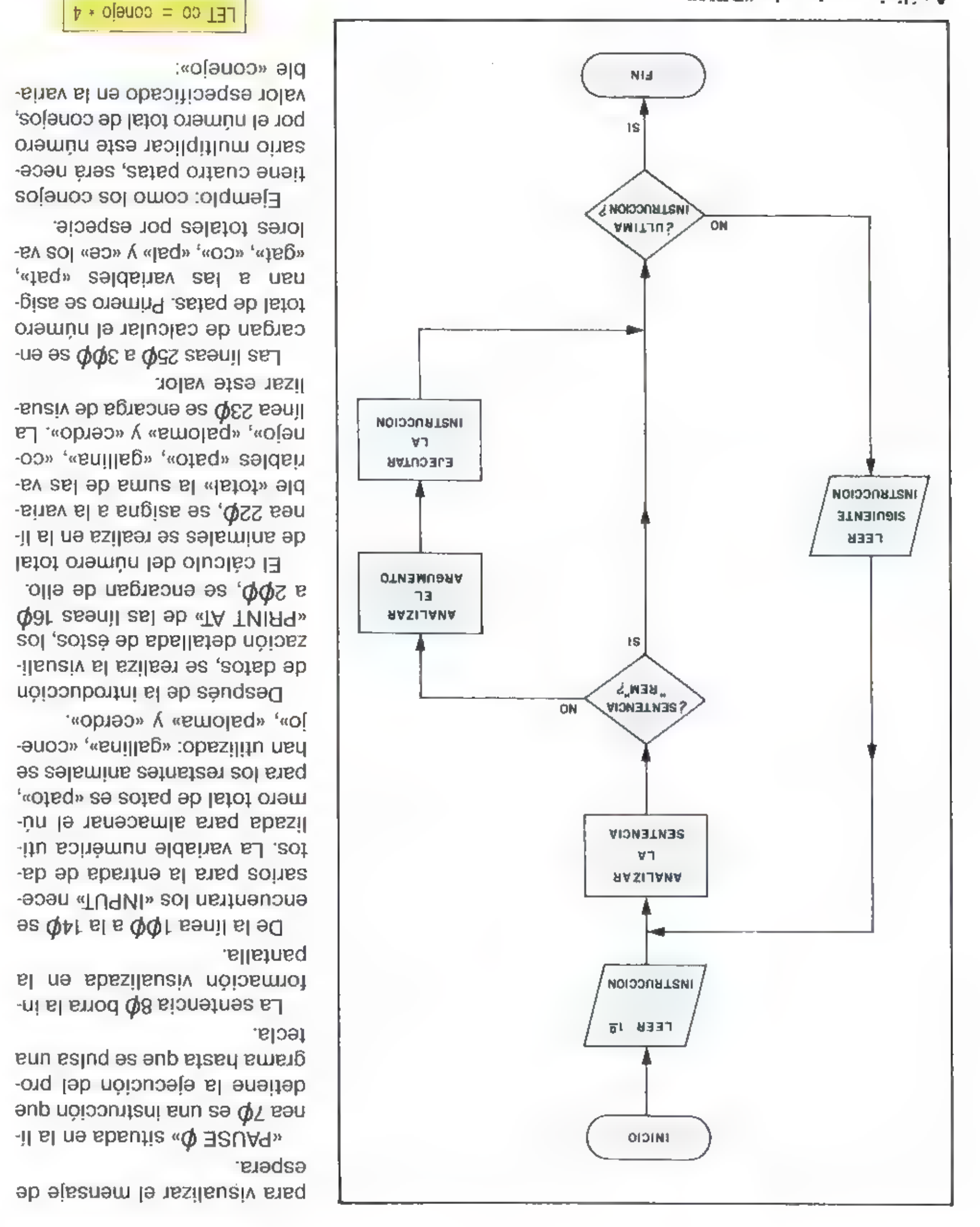
COLOR DEL FONDO (PAPER)

INV. VIDEO

COLOR DEL CARACTER (PAPER)

COLOR DEL FONDO (INK)

Funciones de vídeo.



Análisis sentencia "REM".

blanco. Por último, «CLS» es una sentencia que borra la pantalla y asigna de inmediato los colores especificados en los comandos anteriores.

Las sentencias 4ϕ y 5ϕ son del tipo «PRINT AT» y sirven para dar al usuario una pequeña información sobre la finalidad del programa.

La instrucción 6ϕ utiliza el canal de comunicación ϕ para visualizar el mensaje de espera.

«PAUSE ϕ» situada en la línea 7ϕ es una instrucción que detiene la ejecución del programa hasta que se pulsa una tecla.

La sentencia 8ϕ borra la información visualizada en la pantalla.

De la línea 1ϕϕ a la 14ϕ se encuentran los «INPUT» necesarios para la entrada de datos. La variable numérica utilizada para almacenar el número total de patos es «pato», para los restantes animales se han utilizado: «gallina», «conejo», «paloma» y «cerdo».

Después de la introducción de datos, se realiza la visualización detallada de éstos, los «PRINT AT» de las líneas 16ϕ a 2ϕϕ, se encargan de ello.

El cálculo del número total de animales se realiza en la línea 22ϕ, se asigna a la variable «total» la suma de las variables «pato», «gallina», «conejo», «paloma» y «cerdo». La línea 23ϕ se encarga de visualizar este valor.

Las líneas 25ϕ a 3ϕϕ se encargan de calcular el número total de patas. Primero se asignan a las variables «pat», «gat», «co», «pal» y «ce» los valores totales por especie.

Ejemplo: como los conejos tiene cuatro patas, será necesario multiplicar este número por el número total de conejos, valor especificado en la variable «conejo»:

```
LET co = conejo * 4
```

Posteriormente se asigna a la variable «patas» la suma de las variables «pat», «ga», «co», «pal» y «ce».

Por último el valor de la variable «patas» se visualiza con el «PRINT AT» de la línea 4ϕϕ.

Zonas de visualización.

Programa «ECUACION»

Salvar el programa de la forma:

```
SAVE "ecuación"
```

Este programa calcula las dos raíces de una ecuación de segundo grado del tipo:

$$ax^2 + bx + c = \phi$$

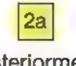
Los dos valores de «x» que cumplen esta ecuación se calculan con la fórmula:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Los valores que deben introducirse para que el programa calcule dichas ecuaciones son los correspondientes a las variables «a», «b» y «c».

Las funciones de las sentencias que componen dicho programa son las siguientes:

1ϕ : Comentario con el nombre del programa.
2ϕ : Asignación de los colores de borde, papel y tinta.
4ϕ — 14ϕ : Primera pantalla de información.
15ϕ — 19ϕ : Segunda pantalla de información.
21ϕ : Introducción de los valores de las variables «a», «b» y «c».
23ϕ — 25ϕ : Visualización detallada de las variables.
27ϕ — 3ϕϕ : Cálculo de las dos raíces. En este programa se ha utilizado la sentencia «SQR» que calcula la raíz cuadrada del argumento que va entre paréntesis. El cálculo se ha realizado en varias etapas, primeramente se han evaluado las partes comunes; a la variable «raíz» se le ha asignado el resultado de:



$$\sqrt{b^2 - 4ac}$$

y a la variable «divisor»



$$2a$$

posteriormente y a partir de estas dos variables se han obtenido las dos



2ϕ — 7ϕ : Entrada y visualización del contenido de las variables «capital», «réditos» y «tiempo».
9ϕ : Cálculo de los intereses. A la variable «interés» se le asigna el resultado de la fórmula.
11ϕ : Visualización de los resultados.

Programa «GRADOS»

Almacenarlo en cinta, por ejemplo, de la forma:

```
SAVE "grados"
```

El programa «GRADOS» consta de dos partes, en la primera transforma un valor de grados *centígrados* (°C), introducido por teclado, en grados *Fahrenheit* (°F) de acuerdo con la fórmula:

$$°F = \frac{9}{5} °C + 32$$

La variable «c» contiene los grados centígrados a transformar y la variable «fahrenheit» el resultado.

En la segunda parte hace la transformación inversa, es decir, transforma un valor de grados Fahrenheit en centígrados, la fórmula implementada en este caso es:

$$°C = (°F - 32) \cdot \frac{5}{9}$$

La variable «f» contiene los grados fahrenheit a transformar y la variable «centígrados» el resultado.

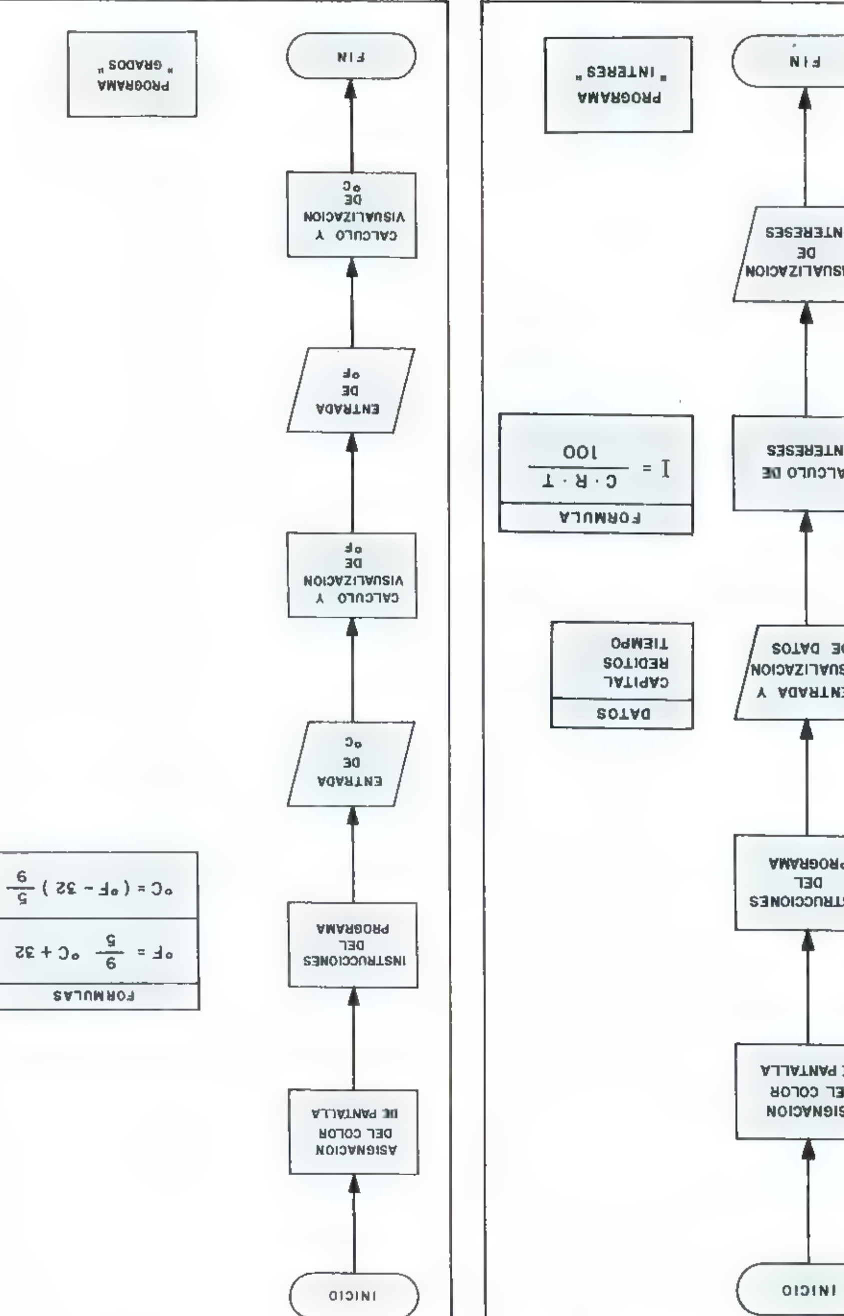
El programa ha sido estructurado de la siguiente manera:

1ϕ : Comentario con el nombre del programa
2ϕ : Asignación del color azul para el borde y el fondo y blanco para los caracteres.
4ϕ — 13ϕ : Breve descripción del programa. En la línea 11ϕ se utiliza el canal de comunicación ϕ.
15ϕ — 16ϕ: Entrada y visualización de la variable «c».
18ϕ — 19ϕ: Cálculo y visualización del resultado en grados Fahrenheit.
21ϕ — 22ϕ: Entrada y visualización de la variable «f».
24ϕ — 25ϕ: Cálculo y visualización del resultado en grados centígrados.

Programa «FICHA»

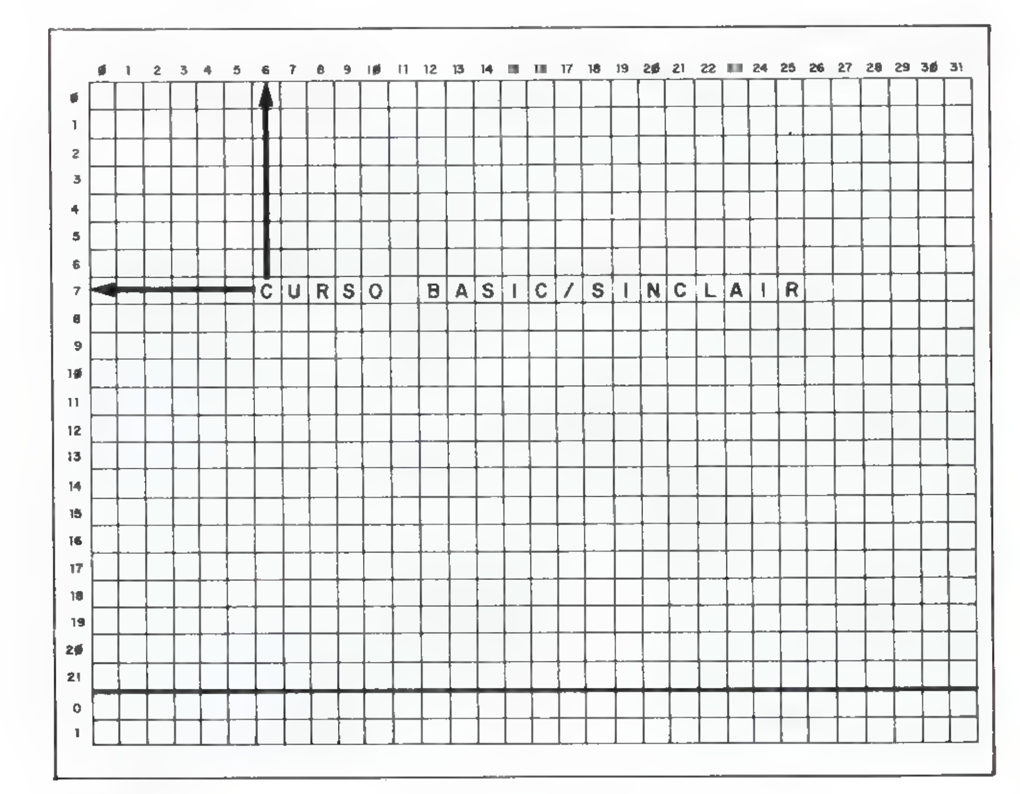
Salvarlo de la forma habitual:

```
SAVE "ficha"
```

Programa "Interés".

Programa "Grados".



Print AT 7,6.

raíces, «raiz 1» y «raiz 2».

41ϕ — 42ϕ : Visualización de los resultados.

Programa «INTERES»

Grábelo en cinta de la forma:

SAVE "interés"

Este programa calcula el *interés simple* de un capital colocado en un banco durante cierto número de años. La fórmula del interés simple implementada en el programa es:

$$I = \frac{C \cdot R \cdot T}{1\phi\phi}$$

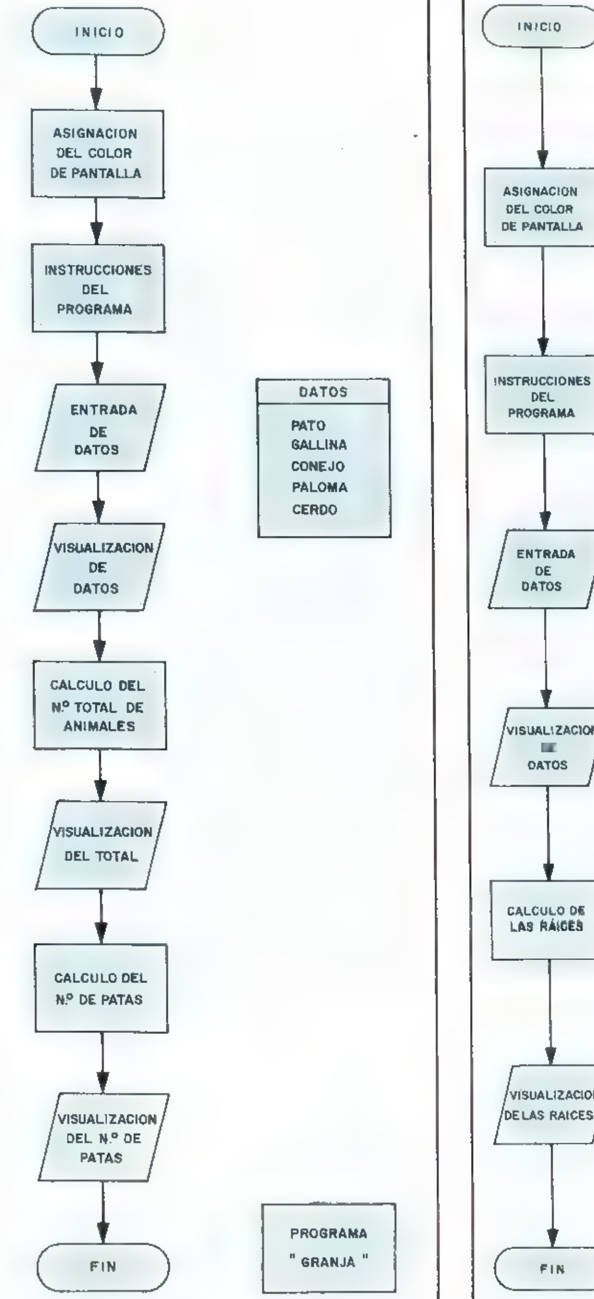
El valor asignado a la variable «capital» debe estar expresado en *pesetas*, el de la variable «reditos» en %, es decir, si el banco proporciona unos intereses al 3%, el valor a introducir deberá ser «3», y por último el asignado a la variable «tiempo» deberá ser expresado en *años*.

La estructura del programa es la siguiente:

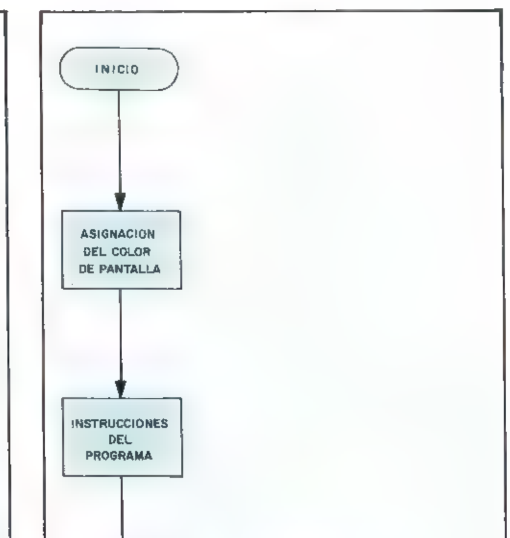
1ϕ : Comentario con el nombre del programa.

11 : Asignación del color azul para el borde y el fondo y blanco para los caracteres.

13 — 18 : Visualización de una breve descripción del programa. La segunda sentencia de la línea 18 es «PAUSE 3ϕϕ», ésta proporciona una temporización de aproximadamente seis segundos, desde que aparece la información hasta que se borra con la siguiente sentencia (CLS). Si durante la temporización se pulsa una tecla, ésta termina y se ejecuta la instrucción siguiente.



Programa "Granja".

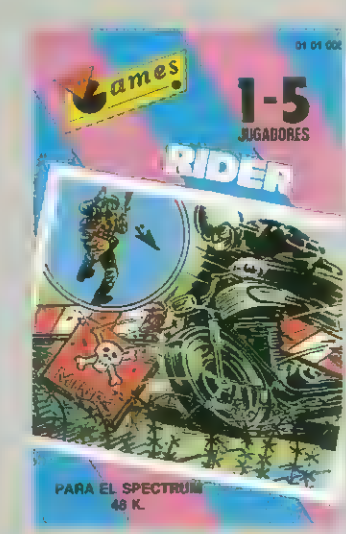


Programa "Ecuación".

llegar a nuestro objetivo. Comienza con un paracaidista que se lanza de un avión y que hay que hacer caer encima de una de las motos que van atravesando la imagen, de izquierda a derecha, ya que de no hacerlo perderíamos el juego y habría que volver a empezar. Y hablando de volver a empezar (y no tiene que ver nada con la película), esto es precisamente lo que tenemos que hacer en innumerables ocasiones, ya que en cuanto cometamos el mínimo fallo, no tendremos más oportunidades para intentar proseguir la misión encomendada en el juego. La única opción en este sentido que nos queda, es la de elegir jugar cinco jugadores, de ese modo, nosotros seremos cada vez uno de los cinco y así se



pude durar un poquito más de tiempo, aunque no será demasiado.
El juego es bastante simplón y, aunque la idea podía haber sido en un principio buena, no se le ha sacado apenas provecho, ni los gráficos, ni el movimiento, ni

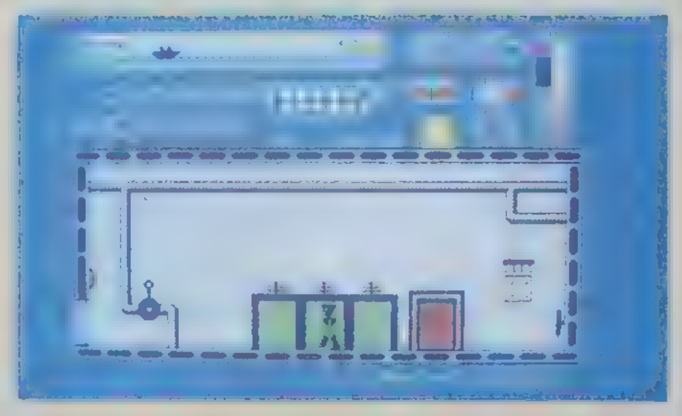


la presentación contribuyen ni mucho menos a alegrarlo.
Los niveles de dificultad, hacen a medida que van siendo más altos, que aumente la velocidad de la moto y el número de minas sea mayor, por lo demás, todo sigue siendo igual de principio a fin.
Se puede realizar dos tipos de movimientos, izquierda y derecha, y acelerar o disminuir la velocidad. Esto último influirá en la maniobrabilidad de nuestro vehízulo, que como es lógico, será más difícil de dominar mientras más rápido vayamos.
Como dijimos al principio es un juego simple, que quizá hoy día se haya quedado algo anticuado, y que por supuesto, está muy lejos de parecerse a algunas de las últimas creaciones que han llegado a nuestras manos.
Un juego en definitiva, con pocas pretensiones.

WORSE SEA

Silversoft.
48 K.
Tipo de juego: Arcade.
PVP.: Sin precisar.

Dirigir a un barco hacia el puerto, puede ser una tarea difícil, para la cual se requieren dotes de navegación lo suficientemente altas como para poder conseguirlo. Pero mucho más difícil aún es hacerlo con un montón de agujeros por los que entra



agua incesantemente, como ocurre en este divertido juego.
El fin no es otro que conseguir llevar al barco al puerto sin que se hunda en el recorrido. Como es lógico, se han producido algunas averías que ocasionan la inundación de las salas.
Nosotros disponemos de un pequeño robot para realizar la tarea de achique, al cual hay que dirigir por el barco en busca de los puntos donde se ha sufrido daños.

Para poder localizarlos disponemos de un mapa-radar donde se pueden apreciar, con detalle, los puntos más críticos. El barco tiene seis salas superiores y cinco inferiores que se comunican con las primeras. Hay que ir con cuidado para lograr evitar que el agua llege hasta los pisos superiores y la inundación sea total.
Para tapar las vías de agua será necesario que hayamos cogido con anterioridad unas placas que se encuentran situadas en cada habitación, con las cuales se pueden tapar dichas vías. Una vez que lo hayamos conseguido, hay que dirigirse hacia la bomba de achique y ponerla en funcionamiento.
Así conseguiremos expulsar el agua de la sala.
El robot necesita energía, y ésta se irá gastando mientras más esfuerzos haga, por eso hay una habitación destinada a recargarle cuando lo necesite. En la parte superior derecha de la imagen, se encuentra un medidor que nos indica la situación actual.
Se trata de un buen programa, que resulta muy entretenido con una buena representación gráfica y una correcta distribución de pantalla.
Un buen juego para entretenernos sin demasiadas complicaciones.

*Ha salido a la venta en Inglaterra, la última creación de la compañía **Ultimate**, que llevará el título de **Alien 8** y, según nos anticipan, se trata de uno de los mejores programas de esta compañía.*

***Sinclair** y la compañía de Software **Activisión** han organizado un concurso de cazadores de fantasmas, en el que ofrecen 100 juegos de recompensa a aquellos que logren cazar el mayor número de ellos. El juego, que ha sido creado por el autor del Decathlon, **David Crane**, está basado en la famosa película **«Los Cazafantasmas»**.*

DE COPAS EN NUEVA YORK

Premiado con 15.000 ptas.

Spectrum 16 K

Javier MELICH MARTARA

El objeto de este original y excepcionalmente bien presentado juego, es coger una serie de objetos situados en la cima de una serie de edificios de distinta altura.

Podemos realizarlo de dos maneras: o bien los recogemos en vuelos rasante desde nuestra nave, por supuesto tenemos una nave (!), o bien rizamos el rizo y nos dejamos caer desde ella hacia la cumbre del rascacielos más cercano.

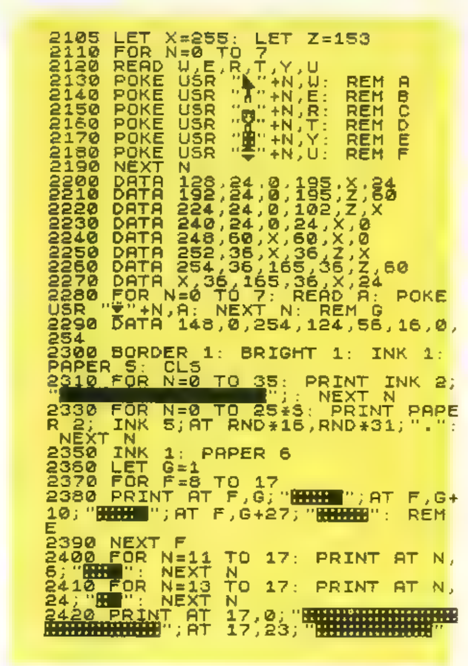
Lamentamos desencantar a todos aquellos que a estas alturas piensen que el juego es demasiado fácil, ya que incorpora a lo largo de toda la zona por la que se mueve nuestra nave una serie de obstáculos que nos destruirán si colisionamos con ellos. El manejo del programa está explicado dentro del mismo; adelantamos sin embargo, que tiene tres niveles de dificultad capaces de «picar» al más pintado.

Suerte y a ver quien coge más copas.

```
NOTAS GRAFICAS
A=◣   B=■   C=■
D=■   E=■   G=■
```

```
  10 GO SUB 2000
  20 LET B=B+1: LET J=J-1: LET T=T-1
  40 PRINT AT 0,8;T;" "
  50 IF T=0 THEN GO TO 800
  60 IF J<=1 THEN LET J=14
  75 IF A>=14 THEN LET A=14
  80 IF B>31 THEN LET B=0
  90 IF INKEY$="0" THEN LET A=A-1
  95 IF INKEY$="O" THEN LET A=A+1
 100 IF INKEY$="Z" THEN GO SUB 300
 105 IF A<=1 THEN LET A=1
 110 IF ATTR (A,B)=113 THEN GO SUB 490
 120 IF ATTR (A,B)=81 THEN GO SUB 500
 130 IF ATTR (A,B)=87 THEN GO SUB 510
 140 IF SCREEN$ (A,B)="." THEN GO SUB 540
 150 IF ATTR (A,B)=84 THEN GO TO 650
 160 PRINT AT A,B;"■"
 170 PRINT AT J,5;"*";AT J,10;"*";AT J,19;"*"
 180 IF SCREEN$ (A,B)="*" THEN GO SUB 600
 200 PRINT OVER 1;AT J,5;"*";AT J,10;"*";AT J,19;"*"
 210 PRINT OVER 1;AT A,B;"■"
 215 IF INKEY$="Z" THEN GO SUB 300
 220 GO TO 20
 305 PRINT AT J,5,"*";AT J,10;"*";AT J,19;"*"
 310 IF J<=1 THEN LET J=14
 320 FOR F=A TO 15
 330 LET T=T-1
 340 PRINT AT 0,8;T;" "
 350 IF T<=0 THEN GO TO 800
 360 IF SCREEN$ (F,B)="*" THEN GO TO 650
 385 IF B>31 THEN LET B=0
 390 IF ATTR (F,B)=113 THEN LET A=1: BEEP .01,0: GO TO 20
 400 IF ATTR (F,B)=81 THEN LET B=B+2: BEEP .01,0
 410 IF ATTR (F,B)=87 THEN GO SUB 510
 420 IF SCREEN$ (F,B)="." THEN PRINT AT F,B;" ": LET R=R-1: PRINT INK 2;AT 0,R;"■": BEEP .1,60: GO TO 580
 430 IF ATTR (F,B)=84 THEN GO SUB 650: LET A=1
 440 PRINT AT F,B;"■"
 441 PRINT AT F,B;"■"
 442 BEEP .005,F+25
 443 PRINT AT F,B;"■"
 444 PRINT AT F,B;"■"
 445 PRINT AT F,B;" "
 450 NEXT F
 460 PRINT OVER 1;AT J,5;"*";AT J,10;"*";AT J,19;"*"
 470 LET J=14
 480 RETURN
 490 BEEP .01,0: LET A=1: RETURN
 500 LET A=A+1: LET B=B+2: RETURN
 510 LET Y=Y+1
 520 PRINT INK 1; PAPER 6; INVERSE 1;AT 17,0;Y
 525 BEEP .01,60
 528 IF Y>=15 THEN GO TO 760
 530 RETURN
 540 LET A=2: LET R=R-1
 550 BEEP .01,1
 560 IF R=15 THEN LET F=1: PRINT PAPER 0; INK 4; FLASH 1;AT 0,12;"◣◣◣": BEEP .5,1
 570 PRINT INK 2;AT 0,R;"■"
 580 IF R<=12 THEN GO TO 800
 590 GO TO 20
 600 BEEP .5,-50
 610 FOR F=0 TO 15
 620 PRINT PAPER 2; INK 5;AT RND*4+2,RND*25+2;"."
 630 NEXT F
 660 FOR F=0 TO 40
 670 LET T=T-1
 680 PRINT AT 0,8;T;" "
 690 PRINT AT 12,30;"■";AT 12,30;"■";AT 12,30;"■"
 700 BEEP .01,-10
 710 IF T<=0 THEN GO TO 800
 720 NEXT F
 730 LET B=1: LET A=2
 740 PRINT ; INK 3; PAPER 1; FLASH 1;AT 12,30;"■"
 750 GO TO 20
 760 POKE 23692,255
 765 FOR N=0 TO 90: PRINT "◣■■■■■";: NEXT N
 770 CLS : PRINT "BONIFICACION 5000 PUNTOS": BEEP 2,-20
 780 LET W=5000
 810 BORDER 0: PAPER 0: CLS
 820 PRINT INK 4;AT 1,15;"◣ COLISIONES ";32-R
 830 PRINT INK 3;AT 3,15;"■ TIEMPO ";T
 840 PRINT INK 6;AT 5,15;"■ COPAS ";Y
 860 PRINT INK 7;AT 7,15;"TOTAL PUNTOS";AT 9,16;(1000*Y)+(100*R)+(5*T)+W
 870 FOR N=20 TO 32-R STEP -1: PRINT INK 4;AT N,1;"◣": NEXT N
 880 FOR N=20 TO 20-(T/25) STEP -1: PRINT INK 3;AT N,3;"■": NEXT N
 890 FOR N=20 TO 20-Y STEP -1: PRINT INK 6;AT N,5;"■": NEXT N
 900 PRINT FLASH 1;AT 20,1;"■■■■■■"
 905 PRINT AT 21,1;"◣ ■ ■"
1070 PRINT AT 14,14;"OTRA PARTIDA (S/N)"
1080 INPUT LINE A$
1090 IF A$="S" THEN RUN
1095 IF A$="N" THEN STOP
1098 IF A$<>"S" OR A$<>"N" THEN GO TO 1080
2000 REM
              INICIO
2005 POKE 23658,8: POKE 23609,15: POKE 23561,15
2010 BORDER 4: INK 7: PAPER 4: CLS
2015 PRINT PAPER 1;"INSTRUCCIONES"
2020 PRINT AT 2,9;"TIENES QUE CAPTURAR LOSOBJETOS QUE  ENCONTRARAS  EN LOS"
2025 PRINT "RASCACIELOS, PERO..."
2030 PRINT "TEN CUIDADO CON EL TIEMPO Y LAS ESTRELLAS"
2050 PRINT AT 10,10;"0  ARRIBA"
2060 PRINT AT 12,10;"O  ABAJO"
2070 PRINT AT 14,10;"Z  DESCENSO"
2080 PRINT INK 0; PAPER 6;AT 20,0;"ESCOGE EL NIVEL
   "'"1=FACIL 2=DIFICIL 3=MUY DIFICIL "
2085 INPUT S
2090 IF S<1 OR S>3 THEN GO TO 2085
2100 REM B C D E A F
```





```
2105 LET X=255: LET Z=153
2110 FOR N=0 TO 7
2120 READ W,E,R,T,Y,U
2130 POKE USR "◣"+N,W: REM A
2140 POKE USR "■"+N,E: REM B
2150 POKE USR "■"+N,R: REM C
2160 POKE USR "■"+N,T: REM D
2170 POKE USR "■"+N,Y: REM E
2180 POKE USR "■"+N,U: REM F
2190 NEXT N
2200 DATA 128,24,0,195,X,24
2210 DATA 192,24,0,195,Z,60
2220 DATA 224,24,0,102,Z,X
2230 DATA 240,24,0,24,X,0
2240 DATA 248,60,X,60,X,0
2250 DATA 252,36,X,36,Z,X
2260 DATA 254,36,165,36,Z,60
2270 DATA X,36,165,36,X,24
2280 FOR N=0 TO 7: READ A: POKE USR "■"+N,A: NEXT N: REM G
2290 DATA 148,0,254,124,56,16,0,254
2300 BORDER 1: BRIGHT 1: INK 1: PAPER 5: CLS
2310 FOR N=0 TO 35: PRINT INK 2;"■■■■■■■■■■■■■■■■";: NEXT N
2330 FOR N=0 TO 25*S: PRINT PAPER 2; INK 5;AT RND*16,RND*31;".": NEXT N
2350 INK 1: PAPER 6
2360 LET G=1
2370 FOR F=8 TO 17
2380 PRINT AT F,G;"■■■■";AT F,G+10;"■■■■";AT F,G+27;"■■■■": REM E
2390 NEXT F
2400 FOR N=11 TO 17: PRINT AT N,6;"■■■": NEXT N
2410 FOR N=13 TO 17: PRINT AT N,24;"■■": NEXT N
2420 PRINT AT 17,0;"■■■■■■■■■■■■■■■■■■■■■■■";AT 17,23;"■■■■■■■■■"
```

```
2425 PRINT INVERSE 1; INK 4; PAPER 2;AT 8,31;"■"
2430 PRINT AT 8,26;"■■■": PRINT PAPER 2; INK 4; INVERSE 1;AT 9,26;"◣ ": REM A
2440 FOR F=0 TO 5
2450 READ N,M
2460 PRINT PAPER 2;AT N,M;"◣"
2470 NEXT F
2480 DATA 8,4,11,8,8,14,17,17,17,27,13,25
2490 INK 6
2500 FOR N=0 TO -31 STEP -1
2510 PLOT 0,31: DRAW 255,N: NEXT N
2600 LET L=15
2610 FOR F=0 TO 15
2620 READ N,M
2630 PRINT PAPER 2; INK 7;AT N,M;"■": REM G
2640 NEXT F
2650 DATA L,0,7,2,L,5,10,7,L,9,L,10
2660 DATA 7,12,L,15,L,L,L,18,L,19,L
2670 DATA 20,L,23,L,26,12,24,7,3■
2680 LET L=16
2690 FOR F=0 TO 10
2700 READ N,M
2710 PRINT PAPER 4; INK 3; FLASH 1;AT N,M;"■"
2720 NEXT F
2730 DATA L,0,L,5,L,9,L,10,L,15
2740 DATA L,L,L,18,L,19,L,20,L,23,L,26
2760 PRINT BRIGHT 0; INK 1;AT 0,0;"■■■■◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣◣"
2770 PRINT INK 2;"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■"
2780 PRINT INK 3; PAPER 0; FLASH 1;AT 12,30;"■"
2790 LET A=1: LET B=0: LET R=32
2800 LET T=500
2810 LET W=0: LET J=14: LET Y=0
2820 PRINT INK 0; PAPER 6;AT 0,0;" TIEMPO "
2830 INK 6: PAPER 2
2900 RETURN
4000 REM
```

MATRICES

Spectrum 16 K

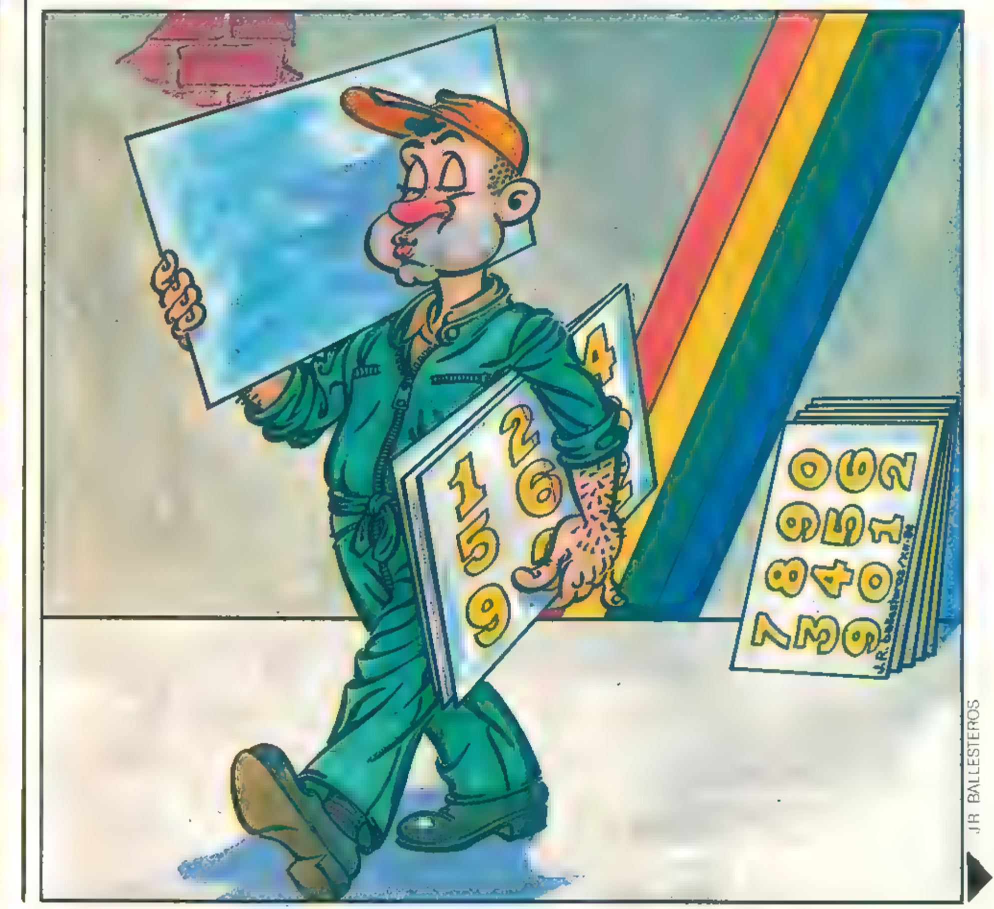
Premiado con 15.000 ptas

José M.ª REUS TERCERO

Estamos ante un programa de utilidades que, como su propio nombres indica, resuelve todo tipo de matrices, es decir, resuelve un sistema de «n» de ecuaciones con «n» de incógnitas, que es el mismo que el grado del determinante.

Concretamente, con este útil programa podrá calcular el valor del determinante, calcular las raíces del sistema de ecuaciones y calcular, por fin, la matriz inversa de la dada.

Si con estas explicaciones no han que-



dado definidas las funciones del programa, les aseguramos que tendrán una visión totalmente clara de él, con tan solo conectar su ordenador. Compruébelo.

```
   4 INPUT "Grado del determinante o numero de ecuaciones del sistema n=";n
   5 DIM a(n,n): DIM o(n)
   7 INPUT "Si desea el valor del determinante c=0 "'"Si desea las raices de un sistema de ecuaciones c=1"'"Si desea la matriz inversa de la dada c=2. Valor de c=";c
   9 LET mc=0
  10 FOR p=1 TO n
  11 FOR q=1 TO n
  12 INPUT ("Matriz, fila(";p;"),columna(";q;"). Valor de a";p;"-";q;"=");a(p,q)
  15 LET mc=mc+a(p,q)*a(p,q)
  16 NEXT q
  17 LET o(p)=p
  18 NEXT p
  20 LET m=SQR mc
  21 FOR s=1 TO n-1
  25 LET amax=ABS a(s,s): LET t=s
  27 FOR p=s+1 TO n
  28 IF ABS a(p,s)>amax THEN LET amax=ABS a(p,s): LET t=p
  29 NEXT p
  30 IF t=s THEN GO TO 40
  32 FOR q=1 TO n
  33 LET b=a(s,q): LET a(s,q)=a(t,q): LET a(t,q)=b
  34 NEXT q
  35 LET b=o(s): LET o(s)=o(t): LET o(t)=b
  40 FOR p=s+1 TO n
  41 LET r=a(p,s)/a(s,s)
  42 FOR q=s TO n
  43 LET a(p,q)=a(p,q)-r*a(s,q)
  44 NEXT q
  46 LET a(p,s)=r
  47 NEXT p
  48 NEXT s
  49 LET d=1
  50 FOR p=1 TO n-1
  51 FOR q=p+1 TO n
  52 IF o(q)<o(p) THEN LET d=-d
  53 NEXT q
  54 NEXT p
  55 FOR p=1 TO n
  56 LET d=d*a(p,p)
  58 NEXT p
  60 IF ABS d<m*1e-7 THEN PRINT "Determinante nulo. Sistema de ecuaciones no determinado": STOP
  61 IF c=0 THEN PRINT "Valor del determinante D=";d: PRINT "Modulo del determinante M=";m
  62 IF c=0 THEN PRINT "Si desea las raices de un sistema de ecuaciones c=1"'"Si desea la matriz inversa de la dada c=2"'"Si no desea nada mas c=0": INPUT "Valor de c=";c
  66 IF c=0 THEN PRINT "Fin del programa": STOP
  67 FOR p=1 TO n-1
  68 LET a(p,p)=1/a(p,p)
  69 FOR q=p+1 TO n
  70 LET b=0
  71 FOR r=p TO q-1
  72 LET b=b-a(p,r)*a(r,q)/a(q,q)
  73 NEXT r
  74 LET a(p,q)=b
  75 NEXT q
  76 NEXT p
  77 LET a(n,n)=1/a(n,n)
  78 DIM x(n): DIM b(n)
  79 IF c=2 THEN GO TO 118
  81 FOR p=1 TO n
  85 INPUT ("Valor del termino independiente de la ecuacion ";p;" b=");x(p)
  86 NEXT p
  87 FOR p=1 TO n
  88 LET q=o(p): LET b(p)=x(q)
  89 NEXT p
  91 CLS
  92 FOR p=2 TO n
  93 LET b=b(p)
  94 FOR q=1 TO p-1
  95 LET b=b-b(q)*a(p,q)
  96 NEXT q
  97 LET b(p)=b
  98 NEXT p
  99 FOR p=n TO 1 STEP -1
 100 LET b=0
 102 FOR q=n TO p STEP -1
 103 LET b=b-b(q)*a(p,q)
 104 NEXT q
 105 LET x(p)=-b
 106 NEXT p
 107 PRINT "Valor del determinante D=";d: PRINT  "Modulo del determinante M=";m
 108 FOR p=1 TO n
 111 PRINT "Incognita(";p;")=";x(p)
 112 NEXT p
 114 PRINT "Si desea un nuevo conjunto de incognitas que satisfagan las mismas ecuaciones, proceda a introducir un nuevo conjunto de terminos independientes con la instrucciones rep=1. Sino lo desea rep=0"
 115 INPUT "rep=";rep
 116 IF rep=1 THEN GO TO 81
 117 PRINT "Si desea la matriz inversa de la dada c=2 ."'"Si no lo desea c=1": INPUT "c=";c: IF c=1 THEN PRINT "Fin del programa": STOP
 118 FOR q=1 TO n-2
 119 LET a(q+1,q)=-a(q+1,q)
 120 FOR p=q+2 TO n
 121 LET b=0
 122 FOR r=q+1 TO p-1
 124 LET b=b-a(p,r)*a(r,q)
 125 NEXT r
 128 LET a(p,q)=b-a(p,q)
 129 NEXT p
 130 LET a(n,n-1)=a(n,n-1)
 131 NEXT q
 132 LET a(n,n-1)=-a(n,n-1)
 133 FOR p=1 TO n
 134 FOR q=1 TO n-1
 135 IF q>=p THEN LET b=a(p,q)
 136 IF q<p THEN LET b=0
 137 FOR r=q+1 TO n
 138 IF r<p THEN GO TO 140
 139 LET b=b+a(p,r)*a(r,q)
 140 NEXT r
 141 LET a(p,q)=b
 143 NEXT q
 144 NEXT p
 145 FOR p=1 TO n
 146 LET q=o(p): LET x(q)=p
 147 NEXT p
 148 PRINT "Matriz inversa"
 149 FOR r=1 TO n
 150 PRINT ""
 151 PRINT "   Fila",r'"Columna"
 154 FOR p=1 TO n
 155 LET q=x(p)
 162 PRINT "     ";p,a(r,q)
 163 NEXT p
 164 NEXT r
 165 PRINT "Valor del determinante D=";d'"Modulo del determinante M=";m
 170 PRINT "Si desea ahora las raices de un sistema de ecuaciones c=1"'"Si no lo desea c=0"
 171 INPUT "Valor de c=";c
 172 IF c=0 THEN PRINT "Fin del programa": STOP
 173 FOR p=1 TO n
 175 INPUT ("Valor del termino independiente de la ecuacion ";p;" b=");x(p)
 176 NEXT p
 180 FOR p=1 TO n
 181 LET q=o(p): LET b(p)=x(q)
 182 NEXT p
 183 FOR p=1 TO n
 185 LET b=0
 186 FOR q=1 TO n
 187 LET b=b+b(q)*a(p,q)
 188 NEXT q
 190 LET x(p)=b
 192 NEXT p
 193 FOR p=1 TO n
 195 PRINT "Incognita p(";p;")=";x(p)
 196 NEXT p
 200 PRINT "Si desea un nuevo conjunto de incognitas que satisfagan las mismas ecuaciones , proceda a introducir un nuevo conjunto de terminos independientes con la instruccion rep=1. Si no lo desea rep=0"
 201 INPUT "rep=";rep
 202 IF rep=1 THEN GO TO 173
 203 IF rep=0 THEN LET c=0: GO TO 172
```

BARON ROJO

Premiado con 15.000 ptas.

Spectrum 16 K

A. MARAÑON

Este programa, por la magia de los bytes, nos traslada hacia atrás en el tiempo y nos coloca en el escenario de la Primera Guerra Mundial; somos ahora, nada menos, que el famoso Ludwig von Richtoffen, as de la aviación alemana y comandante del no menos famoso «circo de Von Richtoffen».

En plena misión de combate, convertidos en Barón Rojo, nos veremos atacados sin tregua por cazas y dirigibles (Zepelines).

Para poder superar nuestro cometido y derribar el mayor número de aviones enemigos en un terreno montañoso y abrupto, hemos de maniobrar hábilmente y evitar, en la medida de lo posible, ser atacados y perecer en el combate.

NOTAS GRAFICAS

A B C D E F G H I J K L

Contamos para manejar nuestro avión con tres mandos:

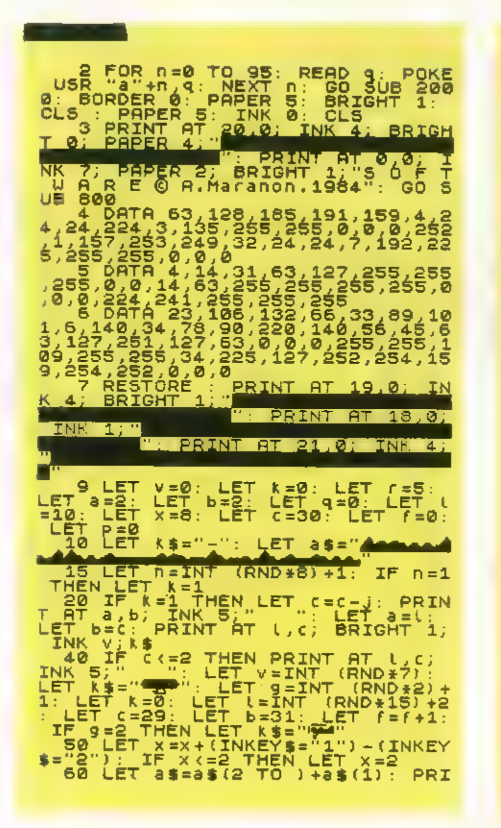
tecla «1», para ir hacia arriba
tecla «2», para ir hacia abajo
tecla «Ø», que nos permitirá efectuar el disparo.

Anímese y conviértase en este famoso personaje que hizo vibrar a sus contemporáneos con sus hazañas.

Para ustedes, ahora esto no será difícil, gracias a su Spectrum y a este programa que le hará pasar muy buenos momentos.

FL. FRONTAN





```
   2 FOR n=0 TO 95: READ q: POKE USR "a"+n,q: NEXT n: GO SUB 2000: BORDER 0: PAPER 5: BRIGHT 1: CLS : PAPER 5: INK 0: CLS
   3 PRINT AT 20,0; INK 4; BRIGHT 0; PAPER 4;"[redacted]": PRINT AT 0,0; INK 7; PAPER 2; BRIGHT 1;"S O F T W A R E © A.Maranon.1984": GO SUB 800
   4 DATA 63,128,185,191,159,4,24,24,224,3,135,255,255,0,0,0,252,1,157,253,249,32,24,24,7,192,225,255,255,0,0,0
   5 DATA 4,14,31,63,127,255,255,255,0,0,14,63,255,255,255,255,0,0,0,224,241,255,255,255
   6 DATA 23,106,132,66,33,89,101,6,140,34,78,90,220,140,56,45,63,127,251,127,63,0,0,0,255,255,109,255,255,34,225,127,252,254,159,254,252,0,0,0
   7 RESTORE : PRINT AT 19,0; INK 4; BRIGHT 1;"[redacted]": PRINT AT 18,0; INK 1;"[redacted]": PRINT AT 21,0; INK 4;"[redacted]"
   9 LET v=0: LET k=0: LET r=5: LET a=2: LET b=2: LET q=0: LET l=10: LET x=8: LET c=30: LET f=0: LET p=0
  10 LET k$="-": LET a$="[redacted]"
  15 LET n=INT (RND*8)+1: IF n=1 THEN LET k=1
  20 IF k=1 THEN LET c=c-j: PRINT AT a,b; INK 5;"  ": LET a=l: LET b=c: PRINT AT l,c; BRIGHT 1; INK v;k$
  40 IF c<=2 THEN PRINT AT l,c; INK 5;"   ": LET v=INT (RND*7): LET k$="": LET g=INT (RND*2)+1: LET k=0: LET l=INT (RND*15)+2: LET c=29: LET b=31: LET f=f+1: IF g=2 THEN LET k$=""
  50 LET x=x+(INKEY$="1")-(INKEY$="2"): IF x<=2 THEN LET x=2
  60 LET a$=a$(2 TO )+a$(1): PRINT AT 17,0; BRIGHT 1; INK 1; PAPER 5;a$
```

```
  62 IF v=5 THEN LET v=7
  65 IF ATTR (x,8)=104 THEN GO TO 90
  70 IF ATTR (l,c-1)=106 THEN PRINT AT r,8; INK 5;"   ": FOR n=1 TO 6: BEEP .02,-20: BORDER INT (RND*7): PRINT AT x,8; INK INT (RND*7); PAPER 5;"": NEXT n: PRINT AT x,8; INK 5;"   ": GO TO 500
  75 IF ATTR (x,8)=105 THEN FOR n=1 TO 5: BEEP .02,-25: PRINT AT x-1,8; INK INT (RND*7); PAPER 5;"": BORDER INT (RND*7): NEXT n: GO TO 500
  90 PRINT AT r,8; INK 5;"  ": LET r=x: PRINT AT x,8; BRIGHT 1; INK 2; PAPER 5;""
 100 IF INKEY$="0" THEN PRINT AT x,10; BRIGHT 1; INK 2; PAPER 5;"............": BEEP .01,10: PRINT AT x,10; INK 5;"            "
 110 IF INKEY$="0" AND l=x AND c<=17 THEN BEEP .03,-30: FOR n=1 TO 8: BORDER INT (RND*7): PRINT AT l,c; BRIGHT 1; INK INT (RND*7); PAPER 5;"": NEXT n: BORDER 0: PRINT AT l,c; INK 5;"    ": LET c=31: LET k=0: LET l=INT (RND*15)+2: LET p=p+1
 120 PRINT AT 1,1;"HUIDOS "+STR$ f;AT 1,16;"DESTRUIDOS "+STR$ p
 125 IF f>=20 THEN GO TO 700
 126 IF p=40 THEN GO TO 900
 130 GO TO 15
 500 PRINT AT 20,0; INK 6; PAPER 1;"INTERCEPTOR DESTRUIDO.  KAPUT.   "
 510 FOR n=1 TO 700: NEXT n: CLS : GO TO 2
 700 FOR n=1 TO 10: BEEP .03,INT (RND*70)-30: NEXT n: PAPER 7: CLS : PRINT AT 10,0; INK 2; BRIGHT 1;"ES USTED UN DESASTRE Y MAS QUE  BARON ROJO SE HA GANADO USTED   A PULSO  EL NOMBRE DE:      N E F A S T O      B A R O N    N E G R O.": PAUSE 0: GO TO 1
 800 INPUT "velocidad enemigo 1 a 5= ";j: IF j<=0 OR j>=6 THEN GO TO 800
 810 RETURN
 900 FOR n=5 TO 25: FOR e=2 TO 10: PRINT AT e,n; INK 2; PAPER 7; INVERSE 1;"*": BEEP .01,10: NEXT e: NEXT n: PRINT AT 6,15; INK 0; PAPER 7; FLASH 1; BRIGHT 1;"X": FOR n=1 TO 100: BEEP INT (RND*3)/(10),INT (RND*35): NEXT n: CLS
 920 FOR n=1 TO 10: BEEP .03,INT (RND*70)-30: NEXT n: PAPER 4: CLS : PRINT AT 10,0; INK 0;"POR TAN MEMORABLE ACCION SE LE  CONDECORA CON LA GRACIOSA GRAN  CRUZ DE LATA RADIACTIVA +": PAUSE 0: GO TO 1
2100 LET d$=" E L  B A R O N  R O J O  "
2200 FOR n=1 TO 60: LET d$=d$(2 TO )+d$(1): PRINT AT 19,0; PAPER 6; BRIGHT 1; INK 2; INVERSE 1;d$: BEEP .01,INT (RND*70)-40: NEXT n: CLS
2300 PRINT AT 2,1; INK 1;"Usted es el baron Von Riftofen,  y su mision es penetrar a la    caza de aviones y dirigibles    enemigos.Si logra llevar a    cabo su mision sera condeco-    rado."
2400 PRINT "Ya sabe Baron, la gloria le es-   pera .SUERTE Y AL TORO."
2450 PRINT : PRINT
2500 PRINT INK 2; BRIGHT 1;"Este es usted  ": PRINT BRIGHT 1; INK 0;"Este es el enemigo  "
2600 PRINT : PRINT "Sus mandos son: 1 - arriba": PRINT "2 - abajo": PRINT : PRINT : PRINT FLASH 1; INK 2; INVERSE 1;"pulse enter para empezar": PAUSE 0: RETURN
```

¡¡GRAN CONCURSO MASTER-MIND!!

UNA INICIATIVA DE MICROWORLD Y MICROHOBBY SEMANAL

Este gran concurso, que por primera vez en España va a enfrentar en competición a programas de ordenador, le brinda la oportunidad de demostrar que es el mejor programando. Y por supuesto, de llevarse grandes premios.
En el concurso participarán programas que jueguen al «Mastermind» (en una modalidad determinada que se explica más adelante), y la forma de seleccionar y elegir al mejor es la competición entre todos. De esta forma, el programa ganador habrá demostrado que es el mejor, al haber superado y eliminado a todos sus contrincantes.

Enviar las cintas a Microworld. Fernández de la Hoz, 64. 28010 Madrid.

El plazo de entrega finaliza el 28 de febrero.

DESCRIPCION GENERAL DEL JUEGO

El juego consiste en que cada programa debe intentar acertar una secuencia de números aleatoria y secreta generada por el otro programa, antes de que el otro programa acierte la secuencia generada por él.

Para intentar conseguirlo, cada uno de los programas irá proponiendo secuencias de números basadas en las «pistas» que el otro programa le vaya dando.

Estas «pistas» estarán referidas a los números que de cada secuencia se vayan acertando, así como a la posición que ocupen dentro de la misma.

Convendremos en que a los aciertos plenos (número y posición), les llamaremos «muertos» (M) y a los aciertos de números sin la posición correcta, les llamaremos «heridos» (H).

Así, si un programa ha conseguido adivinar dos de los números de la secuencia generada por el otro, éste responderá «2H». Pero si uno de ellos está en la posición correcta, entonces deberá responder «1H 1M».

Ganará el programa que consiga acertar primero la secuencia secreta generada por el otro.

En caso de que el programa que empezó primero, acierte antes la secuencia generada por el otro, se le dará a éste una última oportunidad de conseguirlo. Si lo logra, se llegaría a un empate en el juego. Los empates en cada juego, se resolverán mediante un nuevo juego.

REGLAS DEL JUEGO (PROGRAMA)

El programa debe generar una secuencia aleatoria de cinco números, comprendidos entre el 1 y el 9. En esta secuencia no debe haber repeticiones de números, y será secreta para el otro ordenador, pero deberá aparecer en pantalla con el siguiente mensaje:

SECUENCIA GENERADA: nnnnn

A continuación, cada programa debe preguntar quién empieza a jugar primero, con el siguiente mensaje:

COMIENZO YO A JUGAR (S/N): ?

y quedará a la espera de recibir la respuesta, que evidentemente sólo podrá ser una "S" o una "N".

El programa que empiece primero, propondrá una secuencia numérica aleatoria y esperará a que se le introduzca la pista (respuesta) facilitada por el otro programa, así como también la secuencia propuesta por otro programa.

El programa que empezó en segundo lugar, quedará a la espera de recibir la secuencia propuesta por el primero, a la que deberá responder con su pista (respuesta) y su secuencia propuesta, quedando de nuevo a la espera de recibir la pista (respuesta) y la secuencia del que empezó primero.

Este ciclo deberá repetirse hasta que uno de los dos acierte plenamente la secuencia secreta generada por el otro.

El tiempo máximo de respuesta de cada jugada no puede ser superior a 4 minutos.

FORMA DE SELECCION

Los programas admitidos al concurso entrarán en la primera fase del mismo. En esta primera fase se hará competir a los programas en grupos de dos, cargando cada uno de ellos en un ordenador Sinclair ZX Spectrum de 48 K, y jugando una partida. Además cada una de ellas se jugará a dos juegos, comenzando cada vez uno de los dos programas. El programa que pierda los dos juegos quedará eliminado, pasando a la segunda fase el programa que ha ganado los dos. En caso de empate, esto es, si cada uno gana un juego, pasarán ambos a la segunda fase.

En cada partida habrá un operador-árbitro que introduce las jugadas de cada ordenador en el otro. Evidentemente, este operador-árbitro no influye en el juego. Si el autor del programa concursante asiste a la partida, le estará permitido a él mismo introducir la respuesta del otro ordenador en su programa, siempre en presencia del operador-árbitro.



BASES

1. Todos los programas que se presenten deberán «correr» sobre un ordenador Sinclair ZX Spectrum de 48 K.
2. Todos los programas deberán ser originales.
3. Cualquier programa que durante su ejecución, se interrumpa presentando mensajes de error, será automáticamente descalificado.
4. Todos los programas deberán ajustarse a las reglas de juego que aquí se detallan.
5. Los programas deberán enviarse grabados en cassette, con el original por una cara y una copia por la otra.
6. Todas las partidas serán públicas, pudiendo asistir a ellas cuantas personas lo deseen.
7. Tanto el calendario con las partidas a celebrarse como la fecha, lugar y hora de las mismas, se publicarán con la suficiente antelación y siempre, desde las páginas de esta revista.
8. La participación en el concurso supone la aceptación de estas bases, por lo que quedarán automáticamente eliminados aquellos programas que no se ajusten estrictamente a las mismas.
9. No podrán presentarse a este concurso ningún empleado ni familiar de la editorial Hobby Press, ni de la firma Microworld.

PREMIOS

Un capítulo importante de este gran concurso es el de los premios que recibirán los diez primeros finalistas.
En este sentido se distribuirán de la siguiente manera:

- Primer premio: un viaje a Londres para dos personas.
- Segundo premio: un monitor de color.
- Tercer premio: un Spectrum Plus.
- Cuarto, quinto, sexto, séptimo, octavo, noveno y décimo: una serie de lotes de programas de Microparadise y Dinamic.

¡PARTICIPE Y SUERTE!

VICTOR RUIZ UN NOMBRE UNIDO A UNA EMPRESA

Gabriel NIETO

De los programadores españoles que trabajan para el Spectrum, Víctor Ruiz es, sin lugar a dudas, uno de los más prolíferos de todos. Artist, Saimazoom y Babaliba avalan su trayectoria de éxitos.

Víctor, como el resto de los programadores que se dedican actualmente al Spectrum, empezó con un ZX 81. *«Me lo regalaron y casi sin mirar el manual me puse a teclear.»* Sus primeros intentos en este campo pronto empezaron a dar sus frutos, *«lo primero que hice fue un Asteroide, porque el que había para el ZX 81 era bastante malo. Después hice uno de coches y alguno más que dejé sin terminar».*

El principio

En compañía de sus hermanos se puso a trabajar. A todos ellos, como el propio Víctor dice, siempre les ha gustado mucho más programar que jugar, por ese motivo y guiados un poco por algunos programas que les llegaron desde Inglaterra, deciden crear una primera marca que iba a llamarse NCM, para la cual habían preparado un par de programas. Sin embargo, al poco tiempo surgía DINAMIC como un intento de formar un grupo de programadores, a pesar de lo cual nunca pensaron que llegarían a ser una empresa comercial. El primer paso fue poner un anuncio en una revista. Cuando lo hicieron, aún no estaba terminado el Artist. *«Teníamos muchos programas empezados, el Artist, por ejemplo, era sólo un montón de rutinas sueltas, de gráficos. A mí no se me había ocurrido nunca unirlas, pero como ya teníamos el Yehngt empezado y estaba bastante bien, decidí unir todo lo del Artist y al final los acabamos, casi un poco presionados por haber puesto el anuncio.»*

Los primeros pasos de Dinamic son más bien duros, como empresa independiente ellos se lo guisan y ellos se lo comen. *«La idea era hacerlo todo nosotros, vendíamos los programas muy baratos y corríamos nosotros con todos los gastos y con todo el trabajo. Los grabábamos, poníamos las carátulas, nos ocupábamos de la imprenta, todo. Al principio, a pesar de que el anuncio que pusimos era muy malo, tuvimos bastante aceptación.»*

Saimazoon

Saimazoom es el primer gran éxito de Dinamic y de Víctor Ruiz, quien compagina sus estudios con la creación del programa. Al parecer tardó bastante en decidir lo que iba a hacer, *«tenía sólo un mes para hacer el programa. Al principio pensaba hacer una especie de "Pingo", pero finalmente surgió lo de Saimazoon, quizás influenciado un poco por los anuncios de televisión del café, que en aquellos momentos lo pasaban insistentemente. Pensaba hacerle una sola pantalla, pero después vi que no era demasiado liado lo del mapeado y me puse con ello. Diseñé una selva en grande con todos sus detalles y después me dediqué a mapearlo. El personaje de Johny Jones fue lo que más me costó diseñar. Y lo hice conjuntamente con Santi».* El personaje, en un principio, iba a ser un jeep, pero finalmente, se creo este otro que, como dato curioso, hay que decir que se empezó a dibujar por el gorro. El juego logró acabarse en un mes, para lo cual Víctor tuvo que quedarse prácticamente sin dormir más de un día. *«Era la única forma de conseguirlo. Es como más se avanza, aunque te acuestes por la mañana, dedicarle mucho tiempo seguido es el único modo de meterse de lleno en el programa.»*


Nuevos medios para nuevos proyectos.


Víctor en su lugar de trabajo.

Con un Spectrum, un cassette y un televisor en blanco y negro como únicos medios, se obtienen los primeros resultados. El hecho de no disponer de color influye de alguna forma en sus primeras creaciones. Como es lógico, Víctor también se siente preocupado por la protección de sus programas, por eso se pone en contacto con un amigo, Alberto Poveda, y crean la primera rutina de carga rápida, que será, a partir de ese momento, el principal sistema de protección utilizado por Dinamic. Este método fue usado además de como sistema antipirata, como una forma más cómoda de conseguir cargar un programa reduciendo bastante el tiempo de carga.

Con un Spectrum, un cassette y un televisor en B/N se realizan los primeros programas.

Babaliba

Tras Saimazoon llegaría Babaliba, un programa mucho más complejo que el anterior y que además es la segunda parte de éste. El programa se realizó en verano y, en esta ocasión, se cuidaron mucho más los detalles gráficos y el mecanismo general de la aventura. *«Pablo hizo un mapa y nos pusimos a trabajar con él.»*

Víctor, al contrario que otros programadores, ha trabajado casi siempre en equipo y esto es algo que se nota en todos sus programas, donde los detalles se cuidan bastante y la presentación está siempre muy elaborada.

Babaliba es un programa fruto de una estrecha colaboración de los miembros que colaboran en Dinamic, con el estilo incofundible de Víctor Ruiz, que, poco a poco, se va afianzando en nuestro país como una de las más jóvenes promesas en el campo de la programación de juegos, y más en concreto, de videoaventuras que es, sin lugar a dudas, el terreno que mejor domina.

Nuevos proyectos

En la actualidad Víctor trabaja en su nuevo proyecto, Profanation, un programa que empezó su hermano Nacho y para el que se crearon unos gráficos que parece ser superan en mucho a los de los programas anteriores. *«Lo empezó Nacho junto con el Videolimpic, pero luego se quedó un poco colgado, por eso lo he retomado yo porque él tiene ahora otras ideas, y a mí me daba pena desperdiciar todos los gráficos que se habían empezado a diseñar, los cuales están muy elaborados. Vamos a meter además Sprites. Va a ser un juego atómico.»*

Babaliba el último programa.

Además de este proyecto, Víctor prepara una gran sorpresa para las Navidades del 85. Cuando le preguntamos por esto, no quiso adelantarnos nada. *«Va a ser algo totalmente secreto hasta que salga.»*

Dinamic ha evolucionado en este tiempo y, como es lógico, los medios de los que dispone Víctor ahora son también mucho mayores, como sus proyectos, que cada vez son más ambiciosos. En un futuro, incluso, se ha llegado a pensar en la posibilidad de trabajar también para otros ordenadores. *«Estamos metiéndonos con otros ordenadores poco a poco, a ver si sacamos algo para los MSX, aunque aquí en España, todavía no hay muchos, al contrario que en Inglaterra que tienen mucha salida.»*

El personaje de sus juegos está basado en Indiana Jones.

La situación del Software en España es algo que preocupa a todos los programadores, la creación de un mercado potente en nuestro país va siendo poco a poco un hecho, y como era de imaginar, Víctor también opina del tema. *«Hay pocos programadores, pero muy buenos. El Freed y la Pulga son programas que han calado muy hondo en Inglaterra. Nosotros esperamos tener la misma suerte.»*

En Dinamic se cuida mucho la presentación de los programas. Víctor es partidario de dedicar mucho tiempo a confeccionar pantallas, puesto que eso va a influir de alguna forma en la calidad final del producto. La de Babaliba, por ejemplo, tardó en hacerse alrededor de treinta horas, lo que demuestra, sin lugar a dudas, una buena disposición de Víctor a crear productos de calidad. La nueva obra está siendo cuidada al detalle por sus creadores en este sentido. *«La del Profanation es la que más hemos trabajado de todas, espero que sea un golpe para todos. Además, al igual que hicimos con el Babaliba, la vamos a grabar al final del programa para poder sacarla por impresora. Se ha confeccionado sólo en un color y hemos tardado también unas treinta horas en terminarla.»*

Un programador con futuro

En definitiva, Víctor Ruiz es un joven programador con mucho futuro que, a pesar de tener tan sólo 19 años, ya ha incluido su nombre entre los mejores programadores de este país. Su casa de Software favorita es Software Projects, aunque piensa que los programas de Ultimate son ahora mismo los mejores. Su programa inglés favorito es el Knight Lore, mientras que el español es el Freed, al que considera un programa más de su estilo, aunque por supuesto, cuando le preguntamos no pudo olvidarse de «La Pulga». Bebe Coca Cola y come de todo, su grupo de música es Golpes Bajos y le gustan mucho las películas de Indiana Jones, personaje en el cual está basado el héroe de sus programas. Es, en definitiva, un joven programador con mucho futuro por delante, que se ha creado su propio estilo, el cual imprime en todos sus programas. Que su ejemplo sirva para que otros se animen a seguir sus pasos y, poco a poco, lleguemos a crear en España un mercado tan importante como el anglosajón. Material, ganas y buenos programadores hay para ello, además de un público que cada día va tomando más conciencia de cuáles son programas de calidad y cuáles no. Suerte Víctor.

CARGA Y ALMACENAMIENTO DE PROGRAMAS CON VELOCIDAD VARIABLE (I)

Paco MARTIN y José María DIAZ

El bricolaje de los sistemas TURBO. Comandos nuevos y más veloces para su ordenador.

La forma más común utilizada por gran mayoría de los usuarios del ZX Spectrum para grabar y cargar sus programas es la cinta de cassette; este soporte, frente a su bajo costo y relativa fiabilidad, presenta el problema de la lentitud en la transferencia de información.

Por tanto, convendría que el usuario pudiera elegir la velocidad de transferencia casette-ordenador, es decir, SAVE/LOAD programar más deprisa o más despacio que el standard permitido por el propio sistema operativo del ordenador (la ROM).

Antes de entrar en detalles concretos tal vez sería conveniente recordar una serie de conceptos que nos vamos a ver obligados a utilizar a lo largo de este artículo.

La unidad básica del Spectrum para el tratamiento de la información es el «BYTE», es decir, un número binario que varía cíclicamente de Ø a 255; un byte está «compuesto» de 8 bits que tendremos que transferir al cassette o recibir de él.

Existen dos maneras de hacerlo: los ocho bits a la vez (en paralelo) o bit a bit hasta completar el byte de información (en serie).

El Spectrum emplea este último método, por lo que nos centraremos en la transmisión en serie.

Conviene manejar una magnitud que nos mida la velocidad de transferencia de información, para poder manipularla desde un programa; esta magnitud es el «BAUDIO» y representa el tiempo que tarda un BIT en transmitirse. La velocidad «de fábrica» del Spectrum es aproximadamente de unos 1.5ØØ baudios, o sea, 15ØØ/8 BYTES por segundo.

Nosotros hemos realizado un programa en lenguaje máquina cuyo límite inferior ronda los 8ØØ baudios y cuyo límite superior alcanza 5ØØØ.

Es necesario aclarar que no todos los cassettes son iguales ni todos están preparados para recibir/transmitir a muy alta velocidad, así que tendrá que hacer algunas pruebas para encontrar la velocidad que se ajuste a su aparato.

El programa consta de dos grandes partes bien diferenciadas: la primera se encarga de la sintaxis de los nuevos comandos Basic que hemos construido al efecto; la segunda realiza el trabajo de SAVE/LOAD propiamente dicho. Por razones de espacio, nos centraremos en este número en la rutina de sintaxis, y en el próximo en la segunda parte.

Entre los diversos métodos conocidos para ampliar el BASIC del Spectrum, hemos elegido uno que funciona con o sin microdrive; para mayor sencillez, debe correrse la rutina en lenguaje máquina dentro de la propia línea de comandos, bien en modo directo o programa.

Los nuevos comandos tienen una sintaxis muy parecida a la original, excepto por la inclusión de la velocidad en BAUDIOS; por ejemplo, para realizar un SAVE habría que decir:

SAVE BAUDIOS; "NOMBRE" donde BAUDIOS sería un número entre 8ØØ y 5ØØØ inclusive. Esta es la única modificación que hay que incluir en todas las órdenes de SAVE, LOAD, VERIFY y MERGE.

Para poner un ejemplo más concreto, supongamos que nuestra rutina en máquina está ensamblada en la dirección 6ØØØØ y queremos salvar un programa llamado "DEMO" desde el propio programa DEMO a 25ØØ baudios; escribiríamos:

1Ø RANDOMIZE USR 6ØØØØ: REM SAVE 25ØØ; "DEMO"

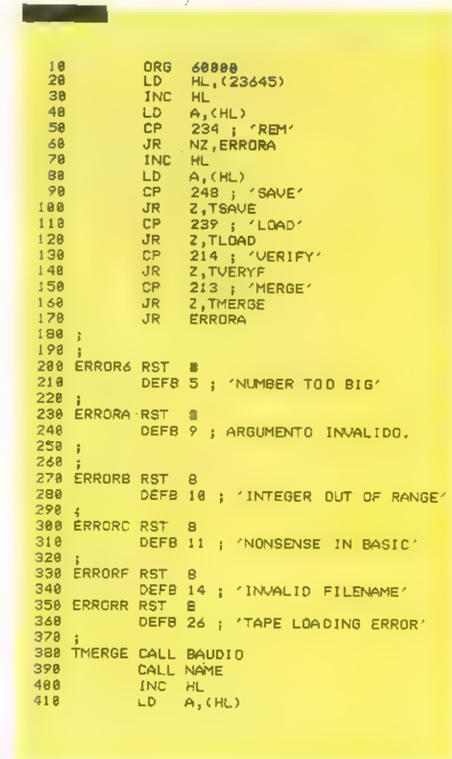
o sin número de línea. El REM es imprescindible ponerlo para que la rutina pueda funcionar.

El procedimiento seguido por la rutina máquina es muy sencillo: existe una variable del sistema localizada en la dirección 23645 cuyo contenido es la dirección del siguiente carácter a interpretar, para ver si es sintácticamente correcto. Tomamos este carácter y los siguientes para ver si responden a la secuencia elegida por nosotros; si es así, la sentencia se ejecuta, si no, mediante la rutina ROM «RST 8» presentamos en pantalla el mensaje de error adecuado.

Como nuestros lectores observarán, la sintaxis de los parámetros que siguen al comando CODE no está «controlada» en todos los casos por razones que se harán claras en los siguientes artículos.

La longitud del listado de esta parte del programa nos ha decidido a emplear el lenguaje ensamblador en aras de la claridad, pensando en aquellos que estén interesados en estudiar la estructura y funcionamiento del programa paso a paso, desarrollo que nosotros, una vez más por razones de espacio, no podemos realizar con el detalle que quisiéramos.

No obstante, publicaremos un programa Basic cargador para los lectores que no dispongan de ensamblador.



```
10          ORG  60000
20          LD   HL,(23645)
30          INC  HL
40          LD   A,(HL)
50          CP   234 ; 'REM'
60          JR   NZ,ERRORA
70          INC  HL
80          LD   A,(HL)
90          CP   248 ; 'SAVE'
100         JR   Z,TSAVE
110         CP   239 ; 'LOAD'
120         JR   Z,TLOAD
130         CP   214 ; 'VERIFY'
140         JR   Z,TVERYF
150         CP   213 ; 'MERGE'
160         JR   Z,TMERGE
170         JR   ERRORA
180 ;
190 ;
200 ERROR6  RST  8
210         DEFB 5 ; 'NUMBER TOO BIG'
220 ;
230 ERRORA  RST  8
240         DEFB 9 ; ARGUMENTO INVALIDO.
250 ;
260 ;
270 ERRORB  RST  8
280         DEFB 10 ; 'INTEGER OUT OF RANGE'
290 ;
300 ERRORC  RST  8
310         DEFB 11 ; 'NONSENSE IN BASIC'
320 ;
330 ERRORF  RST  8
340         DEFB 14 ; 'INVALID FILENAME'
350 ERRORR  RST  8
360         DEFB 26 ; 'TAPE LOADING ERROR'
370 ;
380 TMERGE  CALL BAUDIO
390         CALL NAME
400         INC  HL
410         LD   A,(HL)
```

```
420         CP   13
430         JR   NZ,ERRORC
440         RET
450 ;
460 TVERYF  CALL BAUDIO
470         CALL NAME
480         CALL NSINT
490         RET
500 TSAVE   CALL BAUDIO
510         CALL NAME
520         JR   C,ERRORF
530         CALL SINTAX
540         RET
550 ;
560 TLOAD   CALL BAUDIO
570         CALL NAME
580         CALL NSINT
590         RET
600 ;
610 BAUDIO  CALL NUMERO
620         PUSH HL
630         CP   ";"
640         JP   NZ,ERRORC
650         LD   HL,799
660         SBC  HL,DE
670         JP   NC,ERRORB
680         AND  A
690         LD   HL,5000
700         SBC  HL,DE
710         JP   C,ERROR6
720         LD   DE,(PRES)
730         LD   (BAUD),DE
740         POP  HL
750         INC  HL
760         LD   A,(HL)
770         CP   """
780         JP   NZ,ERRORA
790         INC  HL
800         RET
810 ;
820 NAME    LD   B,10
```

```
830         LD   DE,CABEC+1
840 PNAME   LD   A,(HL)
850         CP   """
860         JR   Z,CMP
870         LD   (DE),A
880         INC  HL
890         INC  DE
900         DJNZ PNAME
910         LD   A,(HL)
920         CP   """
930         RET  Z
940         SCF
950         RET
960 CMP     LD   A,9
970         CP   B
980         LD   A,32
990 LLENA   LD   (DE),A
1000        INC  DE
1010        DJNZ LLENA
1020        RET
1030 ;
1040 SINTAX INC  HL
1050        LD   A,(HL)
1060        CP   13
1070        JR   Z,PPROG
1080        CP   202 ; 'LINE'
1090        JR   Z,LPROG
1100        CP   170 ; 'SCREEN$
1110        JR   Z,SPANT
1120        CP   175 ; 'CODE'
1130        JR   Z,CODE
1140        JP   ERRORC
1150 ;
1160 PPROG  RET
1170 ;
1180 LPROG  RET
1190 ;
1200 SPANT  RET
1210 ;
1220 CODE   CALL NUMERO
1230        LD   (DIRT),DE
1240        CALL COMDAT
1250        LD   A,(HL)
1260        CP   ","
1270        JP   NZ,ERRORC
1280        CALL ENUM
1290        LD   (LONT),DE
1300        CALL COMDAT
1310        RET
1320 ;
1330 COMDAT LD   A,D
1340        OR   E
1350        RET  NZ
1360        DEC  HL
1370        LD   A,(HL)
1380        CP   "0"
1390        JP   NZ,ERRORC
1400        INC  HL
1410        RET
1420 ;
1430 ENUM   CALL NUMERO
1440        CP   13
1450        JP   NZ,ERRORC
1460        RET
1470 ;
1480 NSINT  INC  HL
1490        LD   A,(HL)
1500        CP   13
1510        RET  Z
1520        CP   170 ; 'SCREEN$
1530        JR   Z,SPANT
1540        CP   175 ; 'CODE'
1550        JR   Z,LCODE
1560        JP   ERRORC
1570        RET
1580 ;
1590 LCODE  RET
1600 ;
1610 NUMERO LD   DE,0
1620 CRNUM  INC  HL
1630        LD   A,(HL)
1640        CP   58
1650        RET  NC
1660        CP   48
1670        RET  C
1680        SUB  48
1690        PUSH HL
1700        LD   (PRES),DE
1710        EX   DE,HL
1720        CALL MULT
1730        LD   D,0
1740        LD   E,A
1750        ADD  HL,DE
1760        JP   C,ERRORB
1770        EX   DE,HL
1780        POP  HL
1790        JR   CRNUM
1800 ;
1810 PRES   DEFW 0
1820 ;
1830 MULT   ADD  HL,HL
1840        JP   C,ERRORB
1850        LD   D,H
1860        LD   E,L
1870        ADD  HL,HL
1880        JP   C,ERRORB
1890        ADD  HL,HL
1900        JP   C,ERRORB
1910        ADD  HL,DE
1920        JP   C,ERRORB
1930        RET
1940 ;
1950 BAUD   DEFW 1500
1960 CABEC  DEFB 0
1970 DIRT   DEFW 0
1980 LONT   DEFW 0
```

CONSULTORIO

Conectar al T.V.

Muy Srs. míos.

¿Es verdad que al conectar el ZX-Spectrum a un Televisor de los antiguos, de los de válvulas, éste se estropea?

Gaspar GOMEZ - Huelva

☐ Esto podría ocurrir cuando se conectan aparatos al Spectrum cuyas conexiones no estén debidamente aisladas de la tensión de red, no obstante el Spectrum está aislado por medio del transformador del alimentador. Por tanto una avería por este motivo será bastante improbable, en todo caso vendría ocasionada por otros aparatos conectados al ordenador.

Basic

En el n.º 3 salió en la sección «Curso Basic» que para conectar al ordenador al cassette, éste debía llevar las salidas «Mic» y «Speaker ext». ¿Es la salida «Earphone» o «Headphones» igual a «Speaker ext»?

Juan Diego ALFONSEDA ROJAS
Cartagena

☐ Las indicaciones EARPHONE, EXT SEAKER así como OUTPUT son utilizadas indiscriminadamente por distintos fabricantes para indicar la salida de señal. Lógicamente su utilización es la misma.

Elegir el Spectrum

Pienso comprar un Spectrum, pero no sé la gama que ofrece Sinclair. Por eso les pido, si es posible, que me informen sobre dicha gama. Gracias.

Jesús MUÑOZ - Córdoba

☐ Los modelos actualmente comercializados del Spectrum corresponden a tres versiones del mismo ordenador. Una de ellas con capacidad de 16 K de RAM y las otras dos con 48 K siendo la más reciente de estas el Spectrum +, que incorpora un teclado de mayor calidad (semi-profesional. Para mayor información le aconsejamos se dirija a cualquier establecimiento especializado.

Comunicación entre ordenadores

He oído que algunos ordenadores personales en un futuro próximo podrán comunicarse entre sí.

¿El Spectrum podrá tener esta posibilidad?

Si es así ¿cómo y mediante qué?

Luis Cueto - Málaga

☐ La comunicación entre ordenadores personales se viene realizando desde la comercialización de los Interfaces creados al efecto, entre los cuales el más utilizado es el RS-232 C. Sinclair Research comercializó hace un año aproximadamente, el Interface 1 el cual además de poseer el RS-232 C, contiene una red de comunicación que permite el intercambio de programas y datos con otros Spectrum hasta un total de 64 ordenadores.

Como ya sabe Vd., la sentencia Beep x, y donde y es la nota, y x es la duración de ésta, sirve para producir los sonidos del Spectrum, pues bien, si lo que Vd. desea es que durante la ejecución de un programa al imprimir en la pantalla se produzca un sonido, no tiene más que añadir antes de la orden de impresión, una orden de sonido, por ejemplo:

```
10 beep 0'01, 30:
   Print «Hola»
```

Se producirá un sonido cuanto se imprima en la pantalla Hola.

Si lo que Vd. desea es que se produzca un sonido con cada una de las letras de la palabra, deberá hacer lo siguiente:

```
10 Data «H», «O», «L», «A»
20 For a = 0 to 3 *
30 Read A$
40 Beep 0'1,30
50 Print A$;
60 Next a
```

En el Data deberá estar el texto.

El n.º marcado con * es la cantidad de letras del data menos 1.

Respecto a su sistema de «Reset», no es el más ortodoxo, pero no debe pasarle nada grave.

Sistema MSX

Pienso comprarme un ordenador, para iniciarme, pero me encuentro que en el mercado me aconsejan que espere hasta Enero, porque ha habido una unión de fabricantes que han construido unos modelos en que las cintas y cartuchos se acoplarán unos a otros. Deseo que me informéis sobre esto.

¿Los programas que hacéis para el Spectrum 16 y 48 K, si se copian tal como están escritos, sirven para otros ordenadores?

Un Spectrum 48 K ¿cuántos programas de los que se publican en la revista puede memorizar?

Rosa ALBUIXECHS - Barcelona

☐ Usted se refiere al sistema MSX, que es un intento de standarizar entre otras cosas, el Software, para bajar el precio de éste y de hecho ya en el mercado h y máquinas con este sistema, por ejemplo: Hi Bit, S.V., etc., pero la realidad es que han llegado muy tarde, son más caros y no hay en el mercado la mínima cantidad de Soft para satisfacer al menos exigente, y difícilmente podrán darle la variedad de Software que un Spectrum ofrece.

Los programas que ofrecemos en la revista no pueden ser tecleados directamente en otros ordenadores, puesto que los Basic que se emplean en cada uno, son ligeramente diferentes, igual que la distribución de la pantalla y la memoria.

Aunque se pueden almacenar en la memoria todos los programas que se deseen con tal de que no sobrepasen las 48 K, nuestro consejo para que no tenga problemas es que lo haga de uno en uno.

Compilador, ensamblador, desensamblador

Mis preguntas son las siguientes:

¿Cuáles son, exactamente las misiones del compilador, el ensamblador y el desemsamblador?

Jesús HDEZ AMO - Valladolid

☐ Las misiones de un compilador, ensamblador y desensamblador son las siguientes:

a) Compilador:

El lenguaje Basic es un idioma denominado «interpretado», esto es, la traducción a código máquina antedicha se realiza cada vez que un programa se ejecuta, de forma tal que el ordenador «se olvida» de él cuando termina de ejecutarlo, mientras que un compilador es un programa escrito normalmente en lenguaje máquina, encargado de «traducir», de una vez para siempre, una aplicación escrita en un lenguaje de alto nivel como el Basic, a código máquina puro.

b) Ensamblador:

Los ordenadores sólo comprenden el lenguaje binario, compuesto de unos y ceros; imagine lo tedioso que sería introducir un programa en su ordenador compuesto de series tales como 11100101, y así miles de veces. Para obviar este inconveniente se inventaron los lenguajes de programación, pero su sintaxis se encuentra todavía muy lejos de los números binarios. Así, en una zona intermedia entre unos y ceros y palabras tales como GOTO, GOSUB, etc., se encuentran los ensambladores; éstos son programas que utilizan un lenguaje mucho más cercano al ordenador y se encargan de una serie de tareas tales como la colocación de los bytes del programa en memoria, el cálculo de saltos relativos, el chequeo de errores de sintaxis, etc., de una manera sencilla y rápida; considere la instrucción Basic LET a = 1:

En lenguaje máquina puro, esto sería 00111110, mientras que en lenguaje ensamblador diríamos LD A, o sea, carga (LoaD) el acumulador con un valor determinado; esto último está mucho más cerca de la forma humana de pensar, aunque es bastante más esotérico que la sentencia Basic.

c) Desensamblador:

Su función es convertir series de unos y ceros al lenguaje ensamblador.

Espere nuestras noticias

Les escribo ya que compré el n.º 1 de su revista —grande por cierto— y les mandé la tarjeta de suscripción, de la cual todavía no he tenido contestación.

También les pido si me pueden resolver un problema —creo yo que es un problema— del aparato; al ponerlo en cursor gráfico y al teclear las teclas z y x me aparecen en la pantalla las sentencias PINT y PI, a qué es debido.

Ahora les ofrezco una sugerencia: en los programas de la revista podían introducir programas técnicos.

Les pido el favor que me contesten por carta lo antes posible porque todavía el aparato está en garantía y quiero saber si está estropeado, todo lo demás del aparato está bien.

Esperemos que entre todos hagamos una gran revista que está creciendo semana tras semana.

Ya no me queda más que darles las GRACIAS por adelantado.

Hasta otra, esperando que la próxima sea para colaborar en la revista. Esperando no haberles molestado.

A poder ser contéstenme por carta, lo antes posible. Perdonen mi ansiedad de la contestación.

Se despide con un cordial saludo.

Félix Pablo GRANDE

Publicación de programas

Las preguntas que deseo que me contesten son:

1. ¿Tiene alguna ventaja el suscriptor a la hora de ver publicados sus programas en la revista?

2. ¿Hay mayores probabilidades de que publiquen los programas enviados si se mandan a razón de uno por cinta?

3. ¿Qué tipo de programas desean publicar?

— Juegos (largos o cortos)
— Espectaculares
— Aplicaciones técnicas
— Científicos
— Comerciales
— Aplicaciones a los estudios didácticos
— Subrutinas
— ¿Prefieren el Basic o el Código máquina, o ambos?

4. Los que no tenemos la suerte de contar con una impresora, nos veremos «negros» a la hora de enviar el listado, pues un programa de 100 ó 200 instrucciones se hace interminable de escribir o listar a mano, Teniendo en cuenta que en los programas que se les envíen ustedes podrán acceder fácilmente al listado, ¿es imprescindible enviárselo?

5. En un programa como el que publican en la pág. 11, donde la mayoría de las sentencias son «DATA», ¿podremos suprimir los datos que siguen a este comando «DATA»?; ¿en los programas que usan el código máquina, simplifica esto, la tarea del listado?

M.P.Q. - Pontevedra

☐ Trataremos de responder a sus preguntas por el mismo orden que usted las formula:

1. Las oportunidades de ver publicados sus programas son iguales para todos aquellos que nos los envíen, sean o no suscriptores.

2. Tanto si se nos manda uno o varios programas por cinta, las posibilidades de publicación dependen. exclusivamente, de la calidad de los mismos.

3. Todos los «tipos» de aplicaciones nos interesan por igual, sean Basic o código máquina.

4. No es imprescindible el envío del listado por impresora.

5. Consideramos que suprimir los datos de las sentencias DATA complicaría aún más los programas como el que usted se refiere, ya que quedaría menos claro al lector la lógica del programa.

SI NO QUIERE TECLEAR SUS PROGRAMAS, MICROHOBBY LOS GRABA POR USTED:

(cortar por esta línea)

SOLICITUD DE CINTAS DE PROGRAMAS

Cada mes ponemos a su disposición una cinta con todos los programas publicados en los cuatro números de dicho mes.

Deseo recibir **en mi domicilio** la(s) cinta(s) que indico a continuación, **al precio de 550 pts. cada una,** más 75 pts. de gastos de envío **certificado** por cada cinta solicitada.

- ☐ Números 1 al 4
- ☐ Del 5 al 8
- ☐ Del 9 al 12
- ☐ Del 13 al 16

NOMBRE.. EDAD....................

APELLIDOS..

DOMICILIO..

CIUDAD.................................... PROVINCIA....................

C. POSTAL.................... TELEFONO.................... PROFESION....................

Marco con una (X) en el casillero correspondiente la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre HOBBY PRESS, S.A. ☐ Giro Postal N.º.................... ☐ Contra reembolso

TARJETA DE CREDITO: ☐ VISA N.º.................... ☐ MASTER CHARGE N.º....................

Fecha de caducidad de la tarjeta....................

Firma:

SI NO QUIERE TECLEAR SUS PROGRAMAS, MICROHOBBY LOS GRABA POR USTED:

Franqueo
Postal

HOBBY PRESS, S. A.

Apartado de Correos

n.º **54.062** (Apartados Altos)

MADRID

DE OCASION

• SE VENDE o se cambia por ZX Spectrum 16 ó 48K el lote siguiente: un radio cassette autoreverse «PUNTO AZUL». 10.000 pts. Un maletín de electrónica formado por: soldador, polímetro analógico, cuadro de resistencias (patrón), alicates, destornilladores multiuso, recambios de televisión en general, etc... 15.000 pts. Una carabina de aire «GAMO», con visor telescópico. Un par de bafles de 25W. 3.000 pts. Llamar de 7 a 7,30 tarde al tlf. (981) 32 97 34. Fátima Rodriguez. La Coruña.

• INTERCAMBIO programas de todas clases, tengo una amplia lista. Llamar de 3 a 5 ó de las 10 en adelante al teléfono: 953-69 11 72, mi dirección es: Valentín Alvarez Martínez. C/ Sta. Engracia, n.º11. Linares (Jaén).

• INTERCAMBIO programas para el Spectrum 16/48K. Desearía ponerme en contacto con otros usuarios. Juegos y utilidades. Miguel Angel Gallo. Alminares del Genil. Teléfono: (958) 12 41 59. 18006 Granada.

• INTERCAMBIO todo tipo de programas, a ser posible de utilidades y servicios. Preguntar por: Vicente Paredes Ortiz. C/ Atenas, 19-4.º C Tfno. 675 38 96. Torrejón de Ardoz (Madrid).

• VENDO ZX-81 ampliado a 16K RAM. Manual, cables, juegos, libro de código máquina. 13.000 pts. Juan José Rivero. C/ Orense, 32-5.º 28020 Madrid. Tfno. 455 68 09.

• Hola amigos me gustaría INTERCAMBIAR programas 16/48K para el Spectrum y/o información de toda clase. Mi dirección es: Juan Enrique Dura Oroval. C/ Ramón y Cajal, n.º 83. Carcagente (Valencia). Tel.: 243 36 27.

• INTERCAMBIO, compra y venta de software para el Spectrum de 16/48K. Interesados mandar lista a: Carles Jordi Fernández i San José. Carretera a Bagá, n.º 42-2.º Guardiola de Bergedá (Barcelona).

• INTERESADOS en formar club de software en la provincia de Barcelona. Escribir a: Carles Jordi Fernández I San José. Carretera a Bagá, n.º 42-2.º. Guardiola de Bergueda (Barcelona).

• INTERCAMBIO programas para Spectrum 16 ó 48K. Todos comerciales y clasificados como los mejores del mercado. Mi lista se amplía cada mes más y más. No dejes de llamarme. Carlos González Martínez. C/ Nieves Cano, 65-1.º Izda. Vitoria 01006 (Alava). Telf.: (945) 23 00 69.

• VENDO Spectrum 48K. Todavía en garantía, en 35.000 ptas. (nacional) dispongo también de unos 300 programas que vendo a precio de ocasión. Dirigirse a Marco Ortega Montón. C/ Alonso Allende, 15-7.º D. Portugalete (Vizcaya).

• VENDO Spectrum 16K, con ampliación de 48K, externa. Impresora GP 50 S en garantía. Interface 1 con Microdrive en garantía. Todo por 75.000 pts. También se vende por separado. Regalo 3 cartuchos de Microdrive y más de 50 programas. Pueden dirigirse al teléfono 221 96 11. José Miguel Alba Santamaría.

• INTERCAMBIO toda clase de programas para el ZX Spectrum 48/16 K. Interesados preguntar por Antonio, tfno.: (96) 351 82 76. A partir de las 20-00.

• VENDO interface 1 y un microdrive, 6 cartuchos grabados con 65 programas, y además incluyo el programa «Trans-express» para transferir programas cinta, microdrive y viceversa. Precio a convenir. También vendo 24 números de la revista Ordenador Personal. José Miguel Ródenas Folch . Turo del mar, 1, 6.º 1.ª Montgat (Barcelona). Tfno.: (93) 384 59 46.

• QUISIERA saber si alguien vende ordenadores de 2.ª mano, o cambia cintas. Si es así mándenme información. Mis señas son: Carlos Mereodo Pérez. Avda. Menéndez Pidal. 2. 6 C. Zarzaquemada. Leganés (Madrid).

• DESEARIA contactar con usuarios de ZX Spectrum para intercambio de programas, poseo más de 100. Mando lista a quien la solicite. Dirigirse a: Juan Carlos Claros Trujillo. Grupo Renfe n.º 17. 29007 Málaga. Tfno.: 33 37 26.

• ME GUSTARIA que entre un grupo de amigos formáramos una pequeña sociedad, en la que intercambiaríamos cintas, revistas y todo aquello que nos sirva para conocer mejor nuestro ordenador. Me podéis escribir a: Las Chumberas, Bloque 15. 5.º Izqda. La Laguna (Tenerife). David Castro Piqué.

• INTERCAMBIO programas e información para el ZX Spectrum, preferentemente de 48K. Dirigirse a Angel Manuel Aranguren Reyes. Avda: de los Almendros, 5, 2.º 6. Benidorm (Alicante), o llamar por las tardes a partir de las 6 al teléfono: (965) 85 58 36.

• VENDO ZX-81 nuevo con todos sus accesorios (cables, alimentación, etc), expansión de memoria a 16 K y libros en español e inglés para su uso. Todo el lote por 14.000 pesetas. Llamar al 256 99 43 (tardes). Preguntar por Diego.